University of Alberta

Evaluating DETECT Indices and Item Classification Using Simulated and Real Data that Display both Simple and Complex Structure

by

Xuan Tan

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Measurement, Evaluation and Cognition

Department of Educational Psychology

Edmonton, Alberta
Fall 2006

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-23118-0
Our file Notre référence
ISBN: 978-0-494-23118-0

NOTICE:
The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:
L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

Abstract

Dimensionality assessments are often conducted to validate a construct, which also has implications for diagnostic testing (e.g., Tate, 2002). DETECT is a nonparametric dimensionality assessment procedure with two indices, D_{max} and r_{max}. The indices are used to assess the strength of multidimensionality and whether the dimensional structure identified is simple or complex. DETECT has been shown to work well with test data of simple or approximate simple structure (e.g., Zhang & Stout, 1999b). However, its performance with data of complex structure has only been evaluated in one published study (Gierl, Leighton, & Tan, in press). The present study evaluated the performance of DETECT under conditions of approximate simple and complex structures using simulated and real data. The impact of three factors on the performance of DETECT was investigated—degree of complexity in data structure, correlation between dimensions, and sample size.

In the simulation study, a 3 × 4 × 3 fully crossed design was used. The effect of the three factors on D_{max}, r_{max}, classification accuracy and classification consistency, were studied. Regression analyses for both D_{max} and r_{max}, regressing on classification accuracy, were used to find new critical values for D_{max} and r_{max}. In the real data study, DETECT was used to analyze the SAT 2005 March administration data with hypothesized dimensional structure to confirm results found in the simulation study.

Results from the simulation study suggested that DETECT could adequately identify the dimensional structure of tests (with 80% or higher classification accuracy and consistency) for 15 of 24 cases under the approximate simple structure conditions and 10 of 48 cases under the complex structure conditions. While sample size did not have a

significant effect on DETECT results, the other factors all affected DETECT results significantly. Relaxed evaluation criteria of 0.15 for D_{max} and 0.60 for r_{max} were proposed based on results from the regression analyses. Results from the real data study agreed with the simulation results, and thus indicated the simulated conditions were realistic. Implications to researchers and practitioners were given based on the simulation results. Limitations of the present study and future directions were also discussed.

Acknowledgement

I would like to acknowledge a number of people who have helped me through my doctoral program and the writing of this dissertation. I would like to thank my supervisor, Dr. Mark Gierl. Mark has been a great mentor and an inspiration for me to pursue higher goals in professional development. He has pushed me to come up with research proposals, apply for awards, write for publication, and stick to finishing my final draft before starting to work, and I am deeply indebted to him for that. Without his support, I would not have been able to get this far in three years. I would also like to thank Dr. Todd Rogers and Dr. Mike Carbonaro who are in my supervisory committee. Todd has been a fatherly figure to all the graduate students, and I am not an exception. He went through numerous drafts for my dissertation and made it look much better and more intact. Mike has always been supportive and caring for me through my master's and doctoral programs. He helped me through difficult times and directed me to the area of measurement and evaluation. I will always be grateful for their kindness and support.

My thanks also go to Dr. Connie Varnhagen, Dr. Mazi Shirvani, and Dr. Brian Habing for serving as my committee members and providing helpful comments. Special thanks go to my colleagues in the Centre for Research in Applied Measurement and Evaluation, particularly Ying Cui, Rebecca Gokiert, Changjiang Wang, and Jiawen Zhou, for their helpful discussion. I would also like to thank my graduate school professors, especially Dr. Jacqueline Leighton and Dr. Judy Cameron, whose classes deepened my understanding of the link between measurement and cognitive psychology.

Finally, I would like to thank my family for their continuous support throughout my pursuit of a PHD. I am grateful that my dearest husband, Bihua Xiang, tolerated my

swaying mood and encouraged me all the way through the writing of this dissertation. He sacrificed a lot of his own time to help me format the text and get the copies out. I am also grateful to my parents, Chengquan Tan and Dehua Sun, for their moral support. Every achievement I got, I owe it to them.

Table of Contents

Chapter 1: Introduction 1

Purpose of Current Study 4

Definition of Terms 4

Index of strength of multidimensionality— D_{max} 4

Index of nature of dimensional structure— r_{max} 5

Classification accuracy 5

Classification consistency 5

Organization of the Study 6

Chapter 2: Conceptual Framework and Methods for Dimensionality Assessment 7

Test Dimensionality 7

Unidimensionality and Item Response Theory 8

Multidimensionality and Multidimensional Item Response Theory 12

Methods for Dimensionality Assessment 16

Factor Analytic Methods 17

Conditional Association Methods 19

Geometrical representation of multidimensional items 21

Simple to complex data structures 23

Properties of item conditional covariances 25

DIMTEST 27

HCA/CCPROX 28

DETECT 28

Chapter 3: Review of DETECT 30

Theoretical Development of DETECT 30

Properties of Conditional Covariances 30

Proposal and Refinement of DETECT 34

Evaluation and Application of DETECT 39

Studies Using Data that Focus on Simple or Approximate Simple Dimensional Structures 40

Studies Using Data that Focus on Complex Dimensional Structures 48

Summary 54

Chapter 4: Method 56

Stage 1: Simulation Studies 56

Data 56

LSAT item parameters 58

SAT item parameters 62

Research Design 68

Data Analysis 70

MULTISIM 72

Stage 2: Real Data Studies 73

Data 73

The SAT 73

Participants 74

Samples 74

Data Analyses 74

NOHARM 76

Chapter 5: Results 77

Simulation Results 77

LSAT Results 79

D_{max} and r_{max} indices 79

Classification accuracy (overall) 83

Classification accuracy (dimensions one and two items) 85

Classification accuracy (complex structure items) 86

Classification consistency and misclassification error (overall) 87

Classification consistency and misclassification error (dimensions one and two items and complex items) 92

SAT Results 95

D_{max} and r_{max} indices 95

Classification accuracy (overall) 98

Classification accuracy (dimensions one and two items) 100

Classification accuracy (complex structure items) 101

Classification consistency and misclassification error (overall) 102

Classification consistency and misclassification error (dimensions one and two items and complex items) 106

LSAT Results versus SAT Results 109

Regression Analysis for D_{max} and Classification Accuracy 111

Regression Analysis for r_{max} and Classification Accuracy 114

Real Data Studies 116

Chapter 6: Discussion and Conclusions 122

Restatement of Research Questions and Summary of Methods 122

Discussion of Results and Conclusions 124

Simulation Results 124

Impact of the degree of complexity in data structure 124

Impact of the correlation between dimensions 126

Impact of the sample size 128

Discrepancies between the LSAT and SAT results 128

Adequacy of classification accuracy and consistency 129

Refinement of Evaluation Criteria for D_{max} *and* r_{max} 130

Real Data Results 132

Conclusions 132

Limitations of the Study 133

Implications and Future Directions 134

Educational and Practical Implications 134

Future Research Directions 136

References 138

Appendix A. Visual Basic Code for Batch Processing of Simulation and DETECT Runs 149

Appendix B. Visual Basic Code for Batch Processing DETECT Output and Calculating Classification Accuracy 153

Appendix C. Visual Basic Code for Calculating Classification Consistency 158

List of Tables

Table 1 *Item Parameters for Three Two-dimensional Items* 22

Table 2 *Item Parameters for the Simulated Approximate Simple Structure Items Using the LSAT Parameters* 59

Table 3 *Item Parameters for the Simulated Complex Structure Items Using the LSAT Parameters* 63

Table 4 *Item Parameters for the Simulated Approximate Simple Structure Items Using the SAT Parameters* 65

Table 5 *Item Parameters for the Simulated Complex Structure Items Using the SAT Parameters* 69

Table 6 D_{max} *and* r_{max} *Indices for Simulated Conditions with the LSAT Parameters* .. 80

Table 7 *Overall Classification Accuracy for Simulated Conditions with the LSAT Parameters* 83

Table 8 *Classification Accuracy for Dimensions One and Two for Simulated Conditions with the LSAT Parameters* 86

Table 9 *Classification Accuracy for Complex Structure Items for Simulated Conditions with the LSAT Parameters* 87

Table 10 *Classification Consistency and Misclassification Error Rates between Primary and Cross-Validation Samples for Simulated Conditions with the LSAT Parameters* 89

Table 11 *Classification Consistency and Misclassification Error Rates for Dimensions One and Two for the LSAT Parameters* 93

Table 12 *Classification Consistency and Misclassification Error Rates for Complex Structure Items for Simulated Conditions with the LSAT Parameters* 94

Table 13 *D_{max} and r_{max} Indices for Simulated Conditions with the SAT Parameters* 96

Table 14 *Overall Classification Accuracy for Simulated Conditions with the SAT Parameters* 98

Table 15 *Classification Accuracy for Dimensions One and Two for Simulated Conditions with the SAT Parameters* 100

Table 16 *Classification Accuracy for Complex Structure Items for Simulated Conditions with the SAT Parameters* 101

Table 17 *Classification Consistency and Misclassification Error Rates between Primary and Cross-Validation Samples for Simulated Conditions with the SAT Parameters* 103

Table 18 *Classification Consistency and Misclassification Error Rates for Dimensions One and Two for the SAT Parameters* 107

Table 19 *Classification Consistency and Misclassification Error Rates for Complex Structure Items for Simulated Conditions with the SAT Parameters* 108

Table 20 *Regression Analysis Results for D_{max} and Classification Accuracy* 112

Table 21 *Regression Analysis Results for r_{max} and Classification Accuracy* 115

List of Figures

Figure 1. Vector plot of three two-dimensional items. 23

Figure 2. Vector plot of items showing simple structure. 24

Figure 3. Vector plot of items showing approximate simple structure. 24

Figure 4. Vector plot of items showing complex structure. 25

Figure 5. Vector plot of simulated items for the approximate simple structure condition using the LSAT parameters. 60

Figure 6. Vector plot of simulated items for the complex 40% structure condition using the LSAT parameters. 61

Figure 7. Vector plot of simulated items for the complex 80% structure condition using the LSAT parameters. 62

Figure 8. Vector plot of simulated items for the approximate simple structure condition using the SAT parameters. 66

Figure 9. Vector plot of simulated items for the complex 40% structure condition using the SAT parameters. 67

Figure 10. Vector plot of simulated items for the complex 80% structure condition using the SAT parameters. 68

Figure 11. Linear and nonlinear fitting functions for regressing D_{max} on classification accuracy. 113

Figure 12. Linear and nonlinear fitting functions for regressing r_{max} on classification accuracy. 116

Figure 13. Vector plots of items for the composite tests. 118

Figure 14. Vector plots of items for the Math subtest. 120

Chapter 1: Introduction

In order to identify, interpret, and validate the underlying latent construct measured by a test, dimensionality analyses are often conducted (Ackerman, Gierl, & Walker, 2003; Hambleton & Rovinelli, 1986; McDonald, 2000; Nussbaum, Hamilton, & Snow, 1997). The dimensionality of a test refers to the "minimum number of dimensions or statistical abilities required to fully describe all test-related differences among the examinees in a population" (Tate, 2002, p. 184). For a test that is unidimensional, unidimensional item response theory (IRT) procedures can be used to model the test data. However, when the assumptions for test unidimensionality do not hold, which implies that multiple dimensions exist in a test, multidimensional item response theory (MIRT) model should be used to model the test data. To determine which of these two models should be used, dimensionality analyses should be conducted first.

There are a number of parametric and nonparametric procedures available to assess the dimensionality of a set of data. These procedures identify distinct clusters of test items that represent multidimensional latent traits constituting the underlying construct. The dimensional structure produced can then be interpreted substantively to assign labels and associate meanings with the different dimensions. For example, succinct terms such as "spatial knowledge" can be used to characterize the dimension measured by a set of test items for a specific group of examinees. When the two aspects of dimensionality analyses, statistical and substantive, are successfully connected, information about the dimensional structure of a test can be used to interpret the interaction between the examinees and the items. This interpretation has important implications for using test scores for diagnostic purposes. The identification of distinct

dimensions makes it possible to pinpoint students' deficiencies in areas identified as dimensions. Instead of reporting only the total scores for comparative or selective purposes, testing agencies can provide diagnostic feedback, based on performances on different dimensions (either content-based or cognitive) to students and teachers for remedial instructional activities.

The Dimensionality Evaluation to Enumerate Contributing Traits (DETECT; Kim, 1994; Zhang & Stout, 1999b) is a recently developed nonparametric procedure for identifying the dimensions of a dataset. As a nonparametric procedure, it avoids the need to meet strong assumptions underlying the use of parametric procedures. Further, DETECT does not involve computationally intensive techniques. It is also the first nonparametric procedure that tests for the strength of multidimensionality in a test, estimates the number of dimensions, and identifies the primary dimension measured by each test item (Roussos & Ozbek, 2003). It produces two indices, D_{max} and r_{max}. The D_{max} index indicates the strength of multidimensionality in a test. The r_{max} index indicates whether the classification of items represents simple (r_{max} near one) or complex structure (r_{max} near zero).

When tests of simple or approximate simple structure (a test item measures primarily one dimension) are analyzed, it has been shown that DETECT can adequately identify mutually exclusive, dimensionally homogeneous clusters of items, thereby confirming the dimensional structure of a test in many simulation as well as real data studies (e.g., Gierl, Leighton, & Tan, in press; Zhang & Stout, 1999b). Conversely, when tests of complex structure (a test item might measure multiple dimensions) are analyzed, DETECT has been shown to perform inconsistently across samples in several real data

studies (e.g., Gierl, Tan, & Wang, 2005; Leighton, Gokiert, & Cui, in press). Consequently, the performance of DETECT, particularly under conditions of complex structure, is still not clear, and only one study was found in which this issue was systematically investigated through simulation (Gierl et al., in press).

The successful determination of the number of dimensions underlying a test and the meaningful interpretation of the identified dimensions are dependent upon the consistency and accuracy of the dimensionality assessment procedures. As the true underlying dimensional structure is seldom known in real testing situations, decisions on the dimensional structure of a test rely on cross validation using several samples. Only when consistent results are found across samples can we draw conclusions about the dimensional structure underlying a test with confidence. DETECT has been shown to perform well under restricted conditions with simple and approximate simple structures (Gierl et al., in press; Roussos & Ozbek, 2003; Stout, Habing, Douglas, Kim, Roussos, & Zhang, 1996; Zhang & Stout, 1999b). While the existence of approximate simple structure in test data is known for some large-scale tests such as the Analytic Reasoning and Reading Comprehension subtests of the LSAT, for other large-scale tests such as the Logical Reasoning subtest of the LSAT and the School Achievement Indicators Program (SAIP) science test, the test data display a complex dimensional structure (Leighton, et al., in press; Stout et al., 1996). The Gierl et al. (in press) study yielded significant findings on the performance of DETECT in relation to several factors including the degree of complexity in data structure, correlation between dimensions, and sample size. However, the scope of the factors and conditions within factors considered to date is still limited.

Purpose of Current Study

The purpose of the present study was, therefore, 1) to evaluate systematically the performance of DETECT under conditions of both approximate simple and complex data structures using simulated data; 2) to further investigate the impact of three factors on the performance of DETECT—degree of complexity in data structure, correlation between dimensions, and sample size; and 3) to illustrate the connection between the simulated conditions and real testing situations by using DETECT to analyze data from the SAT 2005 March administration. The research questions addressed in this study included:

1. Are the D_{max} and r_{max} indices, classification accuracy (percentage of items classified into the correct dimensions), and classification consistency (percentage of items classified into the same dimensions across samples) of DETECT influenced by the presence of different degrees of complexity in data structure?
2. Are the D_{max} and r_{max} indices, classification accuracy, and classification consistency of DETECT influenced by the correlations among dimensions?
3. Are the D_{max} and r_{max} indices, classification accuracy, and classification consistency of DETECT influenced by the sample size?
4. Is there a relationship between the D_{max} index and classification accuracy? If there is a relationship, what is the direction of the relationships?
5. Is there a relationship between the r_{max} index and classification accuracy? If there is a relationship, what is the direction of the relationships?

Definition of Terms

Index of strength of multidimensionality—D_{max}. D_{max} is the maximum DETECT value produced by partitioning items on a test into mutually exclusive and dimensionally

homogeneous clusters (see p.34 for the formula). It assesses the amount/strength of multidimensionality in a test, namely the distinctiveness of different dimensions. Tests that are unidimensional should produce a D_{max} of zero; tests that have minor dimensions and are essentially unidimensional should produce a D_{max} value close to zero; tests that are multidimensional should produce a D_{max} value significantly different from zero. According to Kim (1994), a D_{max} value of 0.1 or less indicates essential unidimensionality; a D_{max} value greater than 0.1 and less than or equal to 0.5 indicates weak multidimensionality; a D_{max} value greater than 0.5 and less than or equal to 1 indicates moderate multidimensionality; and a D_{max} value greater than 1 indicates strong multidimensionality.

Index of nature of dimensional structure—r_{max}. The r_{max} index assesses whether the partitioning of items produced by DETECT has a simple or a complex dimensional structure (see p.35 for the formula). This index indicates the nature of the dimensional structure of a test. A r_{max} value greater than or equal to 0.80 indicates simple or approximate simple structure, and a r_{max} value less than 0.80 indicates complex structure.

Classification accuracy. Classification accuracy refers to the rate of accurate classification, namely, the percentage of items on a test accurately partitioned into the dimensions that the items intend to measure. This statistic is obtained based on the true dimensional structure of a test. An item is considered as belonging to the dimension on which it has the highest discrimination parameter.

Classification consistency. Classification consistency refers to the rate of consistent classification, namely, the percentage of items on a test consistently partitioned

into the same dimensions across samples. This statistic is obtained based on agreements among samples.

Organization of the Study

First, the conceptual framework of dimensionality assessment and different methods for dimensionality assessment are described and reviewed in Chapter 2. This review is then followed by an introduction to the DETECT procedure and a review of related literature in Chapter 3. Details on the research design, data simulation, and data analyses of the present study are elaborated in Chapter 4. The results are then reported in Chapter 5. Chapter 6 provides a summary and discussion of the results and proposes new guidelines for using and interpreting the DETECT indices. The conclusions and limitations of the present study are then presented. Implications for practice and suggestions for future research conclude this chapter.

Chapter 2: Conceptual Framework and Methods for Dimensionality Assessment

DETECT, a procedure for exploring the dimensional structure of a test, falls in the conceptual framework of dimensionality assessment. This chapter provides an overview of the concepts related to test dimensionality and methods for assessing test dimensionality. In the first section, the two forms of test dimensionality, unidimensionality and multidimensionality, are introduced together with the assumptions involved and the models used to describe them. The discussions focus mainly on the item response theory (IRT) models because DETECT, although a nonparametric procedure, has its theoretical underpinnings rooted in the IRT framework. The DETECT procedure is based on theories about conditional covariances (Junker, 1993; Zhang & Stout, 1999a). These theories evolved from Stout's (1987) conceptualization of "essential unidimensionality", which are spelled out in the IRT language. Thus, discussions on the IRT models suffice as an introduction to the theoretical background for DETECT. In the second section, methods for dimensionality assessment are discussed as to their mechanism and their strengths and weaknesses.

Test Dimensionality

The identification and interpretation of intended content-based or cognitive dimensions of an assessment instrument provide evidence for construct validity, meaning that a test actually measures what was intended to be measured (Cattell, 1946). The dimensional structure of a test can take one of two forms, either unidimensional or multidimensional. While, traditionally, unidimensionality has been assumed for most standardized achievement or aptitude tests[1], several researchers have argued that the presence of multiple subdomains and skills in a test introduces multidimensionality

[1] Unidimensionality was assumed maybe because we had the capacity to analyze only unidimensional tests.

(Reckase, 1979; Reckase, Ackerman, & Carlson, 1988; Roussos & Ozbek, 2003; Traub & Mclean, 1985; Thurstone, 1947; Yen, 1984, 1985). When more than one dimension can be reliably identified and the scores validly interpreted, diagnostic information on students' strengths and weaknesses can be obtained based on their performances on distinct clusters of items representing different dimensions (Luecht, Gierl, & Huff, 2006; *Standards for Educational and Psychological Testing*, 1999; Tate, 2002, 2004).

Unidimensionality and Item Response Theory

Test unidimensionality is assumed when only one dominant dimension influences test performance and that performance on an item is monotone along the ability scale and independent from performance on another item after conditioned on ability. When a test purports to measure only one attribute or dimension and total scores are used for comparisons across individuals, it is essential to assess the fit of the test data to the unidimensional model. Moreover, the assumptions of unidimensionality need to be evaluated and satisfied before it can be claimed that the unidimensional model provides an adequate fit to the data.

The three assumptions underlying the use of the unidimensional IRT model (the monotone homogeneity model) include unidimensionality, monotonicity, and local independence (LI) (Lord, 1980). To satisfy the unidimensionality assumption, there should exist a unidimensional random variable, θ, which denotes ability, that accounts for all examinee performance. The responses for a test with N items can be denoted by $U = (U_1, U_2, ..., U_N)$. The probability of obtaining a response pattern, $u \in U$, can be expressed using the following formula:

$$P(U=u)=\int_{-\infty}^{\infty}P(U=u\mid\theta)f(\theta)d\theta,$$

where $f(\theta)$ is the density function for θ. Monotonicity means that the probabilistic function, $P(U=u\mid\theta)$, is a non-decreasing function of θ. As ability increases, the probability of a correct response (density) increases as well.

Local independence states that the performance for examinees with the same ability on an item is independent of their performance on any other item. This definition is very stringent and often referred to as strong local independence (SLI). For a test with N items, SLI holds if for all possible response patterns U and all θ,

$$P(U=u\mid\theta)=\prod_{i=1}^{N}P(U_i=u_i\mid\theta),$$

where U_i denotes any possible response to item i, and u_i denotes an incidence of response to item i.

SLI requires that, after the ability is held constant, not only are the covariances between any two items zero, but also that the probability of obtaining a response pattern is the product of all item probabilities. This assumption is complex and difficult to verify statistically. For practical reasons, the traditional definition of SLI is replaced by the definition of weak local independence (WLI), also called pairwise LI (MacDonald, 1979). WLI states that for all item pairs, i and j, and all θ, the conditional covariances are zero, which can be expressed as:

$$Cov(U_i,U_j\mid\theta)=0.$$

WLI can also be expressed using probabilistic terms. That is, a test is said to be weakly locally independent (WLI) if for all item pairs, i and j, and all θ,

$$P(U_i = u_i, U_j = u_j \mid \theta) = P(U_i = u_i \mid \theta)P(U_j = u_j \mid \theta).$$

According to MacDonald (1994), although WLI and SLI are not mathematically equivalent, they are practically equivalent since:

> "we are unlikely to suppose that while every pair of items gives statistically independent responses [conditional on a latent variable], responses to some items [conditional on the latent variable] are dependent on responses to two or more other items." (p. 67)

The use of WLI in place of SLI led to the concept of "essential unidimensionality" proposed by Stout (1987). According to Stout (1987), a test of length N is said to be essentially unidimensional if for all item pairs, *i and j*, and all θ,

$$\frac{1}{N(N-1)} \sum_{1 \le i \ne j \le N} \left| Cov(U_i, U_j \mid \theta) \right| \approx 0.$$

Essential unidimensionality means 1) only one dominant dimension exists for a test and 2) the existence of minor dimensions (traits) common to just a few items should not be counted as dimensions. Several statistical procedures based on the analysis of conditional covariances, such as DIMTEST (Stout, 1987), were developed from the principle of essential unidimensionality for the assessment of dimensionality. By statistically investigating conditional covariances using a valid conditioning variable θ, we can test the tenability of WLI and, in turn, the tenability of test unidimensionality.

The assumption of WLI is intertwined with the assumption of dimensionality. When only one dominant dimension exists in a test, WLI will be achieved. In contrast, when more than one dominant dimension exist on a test, WLI can be achieved only if item responses are independent after conditioning on all contributing latent abilities. Thus, to assess test unidimensionality, we can statistically test the WLI assumption by assuming that a unidimensional random ability underlies the test. The way test

unidimensionality is assessed in programs, such as DIMTEST, is by fitting a unidimensional model to the data and testing whether WLI holds or not. The two concepts, WLI and dimensionality, should never be mixed and used interchangeably.

Satisfying the three assumptions of test unidimensionality must occur before unidimensional IRT models are fit to test data. IRT models use a monotonically increasing function called the item response function (IRF) or item characteristic curve (ICC) to describe the relationship between item performance and examinee ability measured by the test and the characteristics of the item. Three of the most popular unidimensional IRT models are the one-, two-, and three-parameter logistic models. However, it should be noted that these three models are only appropriate for dichotomously-scored items. Only the two-parameter logistic (2PL) model is described here since the present study used a 2PL multidimensional IRT model for simulating data.

The two item parameters in the 2PL model are the item discrimination parameter *a* and the difficulty parameter *b*. The *a*-parameter represents the discrimination parameter of an item in separating students into different ability levels. The *b*-parameter represents the difficulty level of an item, which is equal to the ability estimate for students who have a 0.50 chance of answering the item correctly. The 2PL model can be expressed using the following formula:

$$P(\theta_j) = \frac{1}{1+e^{-Da_i(\theta_j - b_i)}},$$

where $P(\theta_j)$ is the probability that an examinee j with ability θ_j answers item i correctly, a_i is the discrimination parameter for item i, b_i is the difficulty parameter for

item *i*, and *D* (=1.7) is the scaling factor for making the logistic function close to the normal ogive function.

IRT models have several desirable features that have important theoretical as well as practical implications. The item parameter estimates are independent of the group of examinees taking the test (when there is large ability differences between two groups, parameter estimates using different groups could be off, but the shapes of the IRFs would be the same for the two groups). Likewise, the examinee ability estimates are independent from the test items. These two features are often called the invariance property of the item parameter estimates and the examinee ability estimates. When assessing model-data fit, these two features are often tested along with the WLI assumption to see if the data fit the unidimensional IRT model. The standard errors associated with ability estimates obtained using IRT models are known, and they vary across ability levels. These features have many important practical implications. For example, computer adaptive testing (CAT) uses different sets of items to test different students and reports scores as the estimated abilities with certain precision because the item and ability estimates are invariant.

Multidimensionality and Multidimensional Item Response Theory

When the assumptions of unidimensionality do not hold and there is empirical evidence that multiple factors account for differences among students, a test is considered multidimensional. In most dimensionality analyses, only the dominant dimensions are studied while the influence of minor dimensions is ignored. This idea is conceptualized as essential dimensionality, which refers to the minimum number of dominant dimensions

required to satisfy the assumption of WLI after conditioning on all the dominant dimensions (Stout, 1987).

According to Tate (2002), there are two main sources of test multidimensionality: planned content and cognitive structure and unintended sources of multidimensionality. Four aspects should be considered when discussing the associated consequences: validity, reliability, test fairness, and score comparability. Test multidimensionality due to the planned content and cognitive structure could be introduced by the inclusion of various content areas and cognitive thinking levels in the test blueprint during the test development stage. Different item formats tapping different levels of thinking skills, such as multiple-choice versus constructed-response, could also introduce multidimensionality. The multiple dimensions, either content-based or cognitive, if congruent with the test plan, are integral parts of a test that are intended to be measured. Their inclusion should not jeopardize the validity of the total test score, and the use of subscores will lead to valid inferences and provide diagnostic information as to student strengths and weaknesses (Kim & Stout, 1993; Reckase, Ackerman, & Carlson, 1988; Walker & Beretvas, 2003).

The second main source, unintended sources of multidimensionality, includes different kinds of systematic nuisance or construct-irrelevant factors produced by inappropriate test development or administration. For example, if a math test includes several items that require not only math reasoning skills but also vocabulary skill for understanding specific words and students have different combinations of these two skills, the vocabulary skill acts as a construct-irrelevant factor that was not intended to be measured. Nuisance or construct-irrelevant factors can also be introduced by test

speededness, differences in student motivation, test-wiseness, and special item formats such as testlets organized according to reading passages. If the influence of these factors contributes significantly to the test composite score, then the validity and reliability of the total test scores are questionable since unintended dimensions were measured. Test fairness is also a serious concern when nuisance or construct-irrelevant factors contribute to test scores. For example, differential item functioning (DIF) often occurs as a result of the differential ability of different subgroups on the nuisance or construct-irrelevant dimensions even though the subgroups have the same ability on the construct intended to be measured (Shealy & Stout, 1993; Roussos & Stout, 1996; Gierl, 2005). The item bias produced by DIF is one of the threats to test fairness (*Standards for Educational and Psychological Testing*, 1999). For a test that has questionable validity, reliability, and fairness, score comparability cannot be achieved through equating. These concerns do not arise when DIF occurs as a result of item impact, namely the subgroups have different abilities on the construct intended to be measured.

For the current study, which focused on test multidimensionality, only the first source of multidimensionality was considered and studied since the identification of multiple dimensions introduced by planned content and cognitive structure is of most importance for construct validation purposes. The identification of nuisance or construct-irrelevant dimensions, which are the direct cause of DIF, is also important for ensuring the fairness of a test and the cleanness of the construct being measured. However, since DIF is not the outcome of interest for the current study and nuisance or construct-irrelevant dimensions are not a part of the construct intended to be measured, the second source of multidimensionality was not considered.

In order to model multidimensional tests that consist of items measuring different levels of multiple skills, multidimensional item response theory (MIRT) should be used. An extension of unidimensional IRT, MIRT uses probabilistic functions to model the interaction between the probability of a correct response to an item with a set of item characteristics and examinees' abilities on two or more latent traits or dimensions. The assumptions of monotonicity and WLI (as a proxy for LI) need to be satisfied before MIRT models can be used. The monotonicity assumption states that the item response surface is monotonically increasing as the abilities on different dimensions increase. The WLI assumption states that, for groups of examinees with the same abilities on all k dimensions, the conditional covariances for all item pairs, i and j, are zero:

$$Cov(U_i, U_j \mid \theta_1, \theta_2, ..., \theta_k) = 0 .$$

Two types of models are generally used to describe test data that are dichotomously scored—the compensatory model and the noncompensatory model. For the compensatory model (Reckase, 1985), low ability on one dimension can be compensated by high ability on another dimension. For example, on a reading comprehension test that also measures specialized content knowledge, such as football rules, a student who has extensive knowledge on that content area could perform well even though he/she might have poor reading skills. For the noncompensatory model (Sympson, 1978), low ability on one dimension cannot be compensated by high ability on another dimension. An example would be a language test that measures both vocabulary and grammar knowledge. Knowing more words would not help a student perform better on items measuring grammatical knowledge. Although item parameters for the noncompensatory model can be estimated using recently developed Markov Chain Monte

Carlo methods, much research still needs to be conducted to apply and evaluate these methods (Ackerman, Gierl, & Walker, 2003). The compensatory MIRT model, on the other hand, is the more commonly used model, and several computer programs, such as NOHARM and TESTFACT, have been developed for estimating its associated item parameters. Data used in the current study were simulated using the compensatory 2PL MIRT model. The 2PL item response function (IRF) for the compensatory MIRT model can be expressed by the following formula:

$$P_i[U_i = 1 | (\theta_1,...,\theta_k)] = \frac{1}{1+e^{-1.7(a_{i1}\theta_1+a_{i2}\theta_2+...a_{ik}\theta_k+d_i)}},$$

where U_i is the response to item i, $\vec{\theta}^T = (\theta_1,...,\theta_k)$ is the examinee ability vector, $\vec{a}_i^T = (a_{i1},...,a_{ik})$ is the item discrimination vector, d_i is a scalar difficulty parameter, and k is the number of dimensions underlying the test. Unlike the difficulty parameters for the unidimensional IRT model, negative d_is indicate more difficult items while positive d_is indicate easier items.

Methods for Dimensionality Assessment

The use of MIRT to model multidimensional test data is preceded by the determination of the correct number of dimensions and the meaningful interpretation of the identified dimensions (Ackerman, 1994; Nandakumar & Ackerman, 2004). Many parametric and nonparametric statistical procedures haven been developed for assessing the dimensionality of a test such as linear and nonlinear factor analysis, residual analysis, the Bejar (1980) analysis method, and methods based on conditional associations (e.g., Hambleton & Rovinelli, 1986; Hattie, 1985; Stout, 2002). Unfortunately, however, no

standard set of recommendations or guidelines exist on the proper use of these procedures for large-scale testing (Tate, 2002).

Methods commonly used today for dimensionality assessment can be classified into two categories, parametric factor analytic methods and nonparametric methods based on conditional associations. The factor analytic methods use the matrix of item correlations (phi or tetrachoric correlations) to extract factors common to clusters of items. Methods based on conditional associations use the matrix of item conditional covariances to partition items into mutually exclusive and dimensionally homogeneous clusters and test the violation of the WLI assumption with the identified item clusters. Both sets of methods have their strengths and weaknesses.

Factor Analytic Methods

Factor analytic methods attempt to model examinee responses to dichotomously-scored test items using either a classical linear model or a nonlinear model. Classical linear factor analysis models the relationship between the item responses and a set of multiple factors or abilities; nonlinear factor analysis models the relationship between the probability of a correct response and a set of multiple factors or abilities. Classical linear factor analysis can be conducted with various factor extraction methods, rules for identifying the number of factors, and rotation/transformation methods, as implemented in SPSS (exploratory, SPSS Inc., 2005) and LISREL (confirmatory, Jöreskog & Sörbom, 1993). The normal-ogive harmonic analysis robust method (NOHARM) (McDonald, 1967, 1997; Fraser & McDonald, 1988) and TESTFACT programs (Bock, Gibbons, Schilling, Muraki, Wilson, & Wood, 1999) use the nonlinear factor analysis model. The nonlinear factor analysis model has been shown to be essentially equivalent to the MIRT

model (McDonald, 1967, 1999). Thus, the item discrimination, item difficulty, and guessing parameters associated with the MIRT model can be estimated using the TESTFACT and NOHARM programs.

Factor analytic methods can be used in two modes, exploratory and confirmatory. When there is no prior belief or strong theoretical support for hypothesis about a test's dimensionality, the exploratory mode should be adopted. First, different numbers of dimensions are specified to be extracted from the correlation matrix. The fit is then assessed by examining the residual correlation matrix or a summary of this matrix such as the root mean square of residuals. The residual matrix is calculated as the matrix of difference between the observed correlation matrix and the correlation matrix reproduced from the number of factors extracted from the observed correlation matrix. Determination of the final number of dimensions is a balanced decision between the parsimony and interpretability of the factor pattern matrix and the fit indices. A factor or pattern matrix that has higher values of residuals may be selected as the final solution if it represents a more parsimonious and interpretable solution with a small number of dimensions.

The confirmatory mode should be adopted when there is strong theoretical support for a test's dimensional structure, either from substantive reviews or from careful test development based on an established content/cognitive structure, pilot studies, or cross-validation studies. In this case, a hypothesized model of the test can be provided by specifying the number of factors and the factor(s) each item loads on. Different fit indices, such as the root mean square residual (RMSR, Fraser, 1988), can be used to assess how well the hypothesized model fits the data, and when the model fails to fit, to provide information on where the model has failed.

Factor analytic methods share the common pitfalls of all parametric procedures. Assumptions made about an assumed model cannot always be satisfied. For example, the multivariate normality assumption in the normal ogive model may not be satisfied. Also it may be problematic to assume that there is a bivariate normally distributed latent response variable underlying the dichotomously-scored items when tetrachoric correlations are used. Classical linear factor analysis is inappropriate for use with dichotomously-scored items even when tetrachoric correlations are used (Hattie, 1985). Research has indicated that increasing nonnormality leads to attenuated item loading estimates and lack of fit (Curran, West, & Finch, 1996; Olsson, 1979). Another problem associated with factor analytic methods is that the requirement of a positive semidefinite correlation matrix is not always satisfied with real data. When a dataset is found not to be positive semidefinite, researchers usually inspect the response data and eliminate redundant variables (Todd Rogers, personal communication, May 25, 2006). The last limitation of factor analytic methods is the indeterminacy of different decision rules for identifying the number of meaningful factors (Mislevy, 1986). The decision rules seem to perform differentially under different circumstances (e.g., high versus low saturation—magnitude of factor loadings, see Hakstian, Rogers, & Cattell, 1982) producing solutions with varying numbers of factors (Zwick & Velicer, 1982, 1986; Leighton et al., in press).

Conditional Association Methods

Methods based on conditional associations are derived either from the WLI assumption of test unidimensionality or from the concept of essential unidimensionality. By testing whether the individual conditional covariances for all item pairs or the sum of the absolute values of the conditional covariances for all item pairs after conditioning on

a single ability are close to zero, one can infer the dimensional structure of a test. Therefore, there are two categories of methods based on conditional associations. The first is based on the review of individual conditional covariances for all item pairs (derived from the WLI assumption). The second is based on a global measure of all conditional covariances (derived from essential unidimensionality). Seven measures of local item dependency based on individual conditional covariances are provided by the IRTNEW program (Chen, 1993; Chen & Thissen, 1997). These indices are all parametric since the conditioning is based on a unidimensional IRT model. These individual-index based measures suffer not only from the weaknesses of parametric procedures as mentioned previously but also from the inflated family-wise error when an omnibus test of conditional covariances for all item pairs is conducted. Furthermore, these procedures do not provide an estimate of the number of dimensions underlying a test, but are useful for identifying local item dependency for selected item pairs in an exploratory mode.

Three commonly used programs based on the global measure of conditional covariances include the dimensionality test (DIMTEST, Froelich & Habing, 2001), hierarchical cluster analysis (HCA/CCPROX, Roussos, 1995), and DETECT (Zhang & Stout, 1999b). These methods have been reviewed and evaluated in many simulation and real data studies (e.g., Hattie, Krakowski, & Swaminathan, 1996; Nandakumar, 1991, 1993; Nandakumar & Ackerman, 2004; Stout et al., 1996; Zhang & Stout, 1999b). They are nonparametric in the sense that number-correct scores are used as the conditioning variable representing the composite ability. As nonparametric procedures, these global methods are not restricted by any model assumptions and are computationally efficient.

To understand how these methods based on global measures work, it is necessary to first understand the properties of item pair conditional covariances. A better way to illustrate the properties of item pair conditional covariances is through geometrical representation of multidimensional items.

Geometrical representation of multidimensional items. A geometrical representation of items in the multidimensional space is often used to help make it easier to understand the relationship between individual items and the multiple dimensions being measured. For illustrative simplicity, only the two-dimensional case is used to explain how geometrical representations are created. The two-dimensional space is represented by a Cartesian coordinate system with the x axis being ability dimension one, θ_1, and the y axis being ability dimension two, θ_2. The origin of the coordinate system represents the population means of abilities on both dimensions. An item is represented by a vector, which, when extended, passes through the origin. The length of the vector, which represents the multidimensional discrimination parameter denoted by $MDISC$, equals $\sqrt{(a_{i1}^2 + a_{i2}^2)}$, where a_{i1} and a_{i2} are the discrimination parameters associated with the two dimensions. The direction of the vector represents the composite of θ_1 and θ_2 at which the item is most discriminating, and is called the direction of best measurement of an item. The direction is given by the angle of the item vector, with respect to the x axis, α_i: $\alpha_i = \arccos(\frac{a_{i1}}{MDISC})$. Thus, items measuring more of dimension one will have an angular direction smaller than 45°, and items measuring more of dimension two will have an angular direction greater than 45°. The location of the vector, the signed distance of the item vector from the origin, represents the multidimensional difficulty parameter,

which equals the multidimensional ability at which the probability of a correct response is 0.5. The signed distance means that a positive or negative sign is associated with the distance measure: item vectors in the first quadrant are given a positive sign; item vectors in the third quadrant are given a negative sign. This multidimensional difficulty parameter is denoted by $D_i = \frac{-d_i}{MDISC}$, where d_i is the scalar difficulty parameter for item *i* as defined in the MIRT model. Since the multidimensional discrimination parameters are always positive, the item vectors lie only in the first and/or the third quadrants. Items lying in the first quadrant represent harder items than items lying in the third quadrant. Figure 1 is an example of the geometrical representation of three items in the two-dimensional space. The item parameters for the three items are included in Table 1.

Table 1

Item Parameters for Three Two-dimensional Items

Items	a_{i1}	a_{i2}	d_i
1	0.75	0.15	-1
2	0.10	0.45	-0.5
3	0.30	0.35	0.5

As illustrated in Figure 1, item 1 has the longest item vector since its multidimensional discrimination parameter is the highest while item 3 lies in the third quadrant since it is an easier item with a positive d_i. Item 1 measures primarily dimension one, as illustrated with a small slope; item 2 measures primarily dimension two with a large slope; and item 3 measures both dimensions comparably with a medium slope.

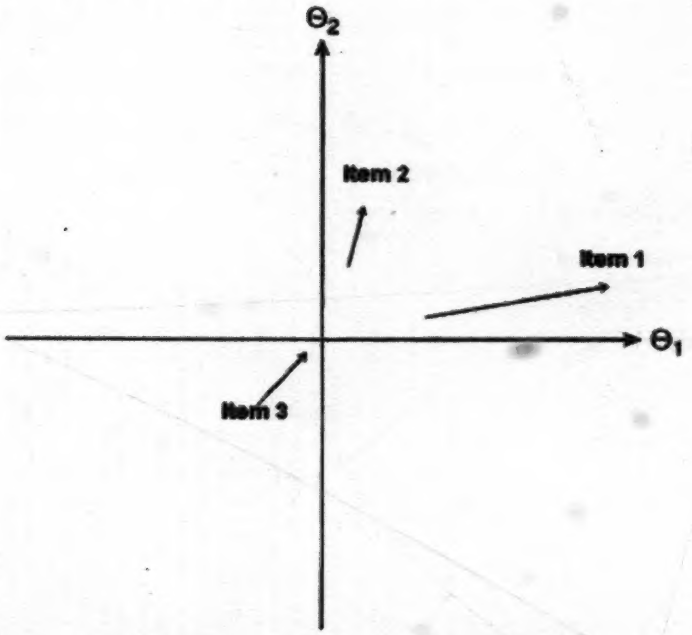


Figure 1. Vector plot of three two-dimensional items.

Simple to complex data structures. The way item vectors cluster in the two-dimensional space relative to the two axes provides useful information about the dimensional structure underlying a test. If all the item vectors of a test lie exactly along the two axes, as illustrated in Figure 2, then the test is considered to have simple structure. Having simple structure means all items in the test measures only one of the dimensions even though the dimensions may be correlated. If the item vectors of a test lie in two narrow sectors close to the two axes as in Figure 3, then the test is considered to have approximate simple structure (Stout, 1987). In this case, the items measure primarily one of the dimensions and slightly the other dimension. As shown in Figure 3, items 1, 2 and 6 measure primarily dimension one, and items 3, 4 and 5 measure

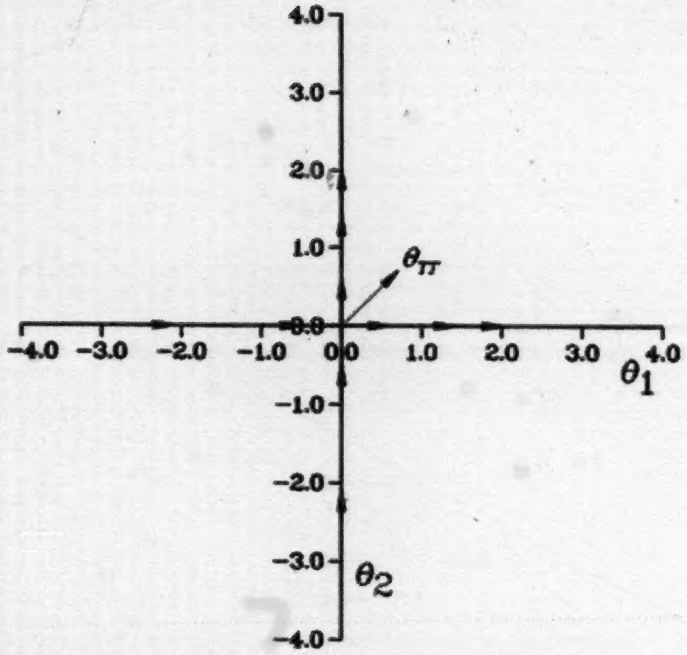


Figure 2. Vector plot of items showing simple structure.

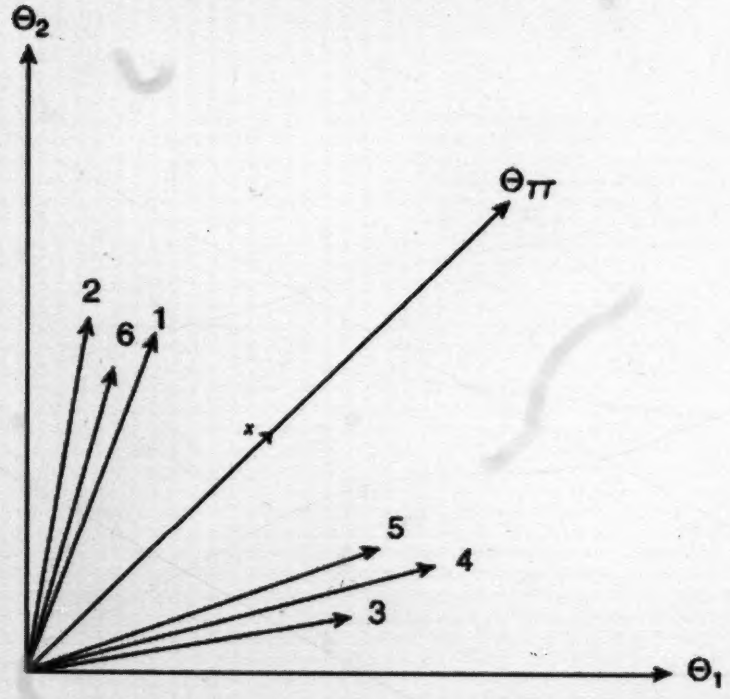


Figure 3. Vector plot of items showing approximate simple structure.

primarily dimension two. If the item vectors of a test spread throughout the first and/or the third quadrants as in Figure 4, then the test is considered to have complex structure. There are items measuring primarily one of the dimensions (items 1 and 2 measure primarily dimension one, and items 6 and 7 measure primarily dimension 2) and also items measuring a composite of the two dimensions with differential weights (items 3 to 5 and items 8 to 10).

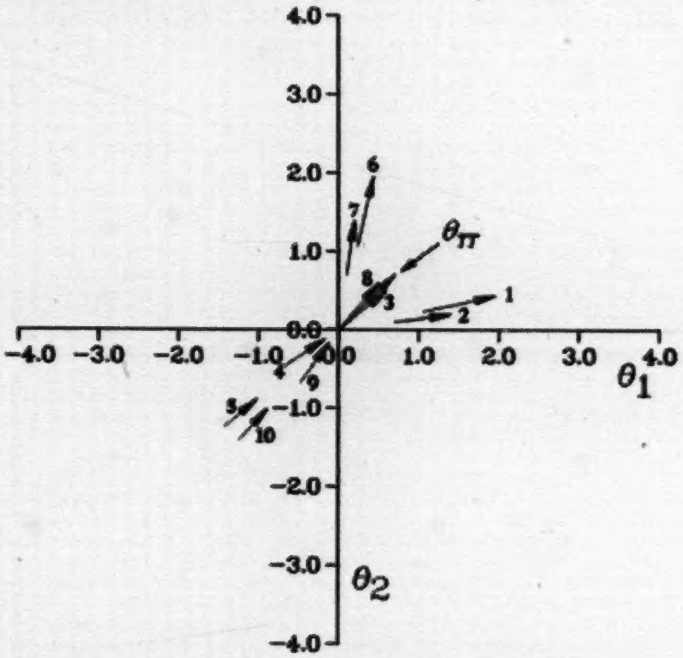


Figure 4. Vector plot of items showing complex structure.

Properties of item conditional covariances. Zhang and Stout (1999a) established the theoretical basis for using conditional covariances to determine the dimensional structure of a test. The conditional covariances have been shown to exhibit a consistent sign behavior when the conditioning variable is set to be the composite ability represented by the vector θ_{TT} in Figures 2, 3, and 4. θ_{TT} is defined to be a standardized linear combination of the examinee ability vector $\vec{\theta}$:

$$\theta_{TT} = \vec{\alpha}^T \vec{\theta} = \sum_{i=1}^{k} \alpha_i \theta_i ,$$

where k is the number of dimensions underlying a test, $\vec{\alpha}^T = (\alpha_1, \alpha_2, ..., \alpha_k)$ is the direction of best measurement for the composite ability with α_i equal to $\frac{a_i}{\sqrt{a_1^2 + a_2^2 + ... + a_k^2}}$, and $\vec{\theta}^T = (\theta_1, \theta_2, ..., \theta_k)$ is the examinee ability vector. The conditional covariance for an item pair, *i* and *j*, is positive if items *i* and *j* measure similar ability dimensions, negative if items *i* and *j* measure different ability dimensions, and zero if one of the items measures the composite ability θ_{TT}. Illustrated graphically, in the two-dimensional case shown in Figure 3, the conditional covariance of an item pair is positive if the item vectors in the pair lie on the same side of the vector θ_{TT} representing the direction of best measurement for the conditioning variable, negative if the item vectors in the pair lie on the opposite side of θ_{TT}, and close to zero if one of the item vectors lies near θ_{TT}. For example, the conditional covariance is positive for items 2 and 6 and negative for items 2 and 5. An item vector *x* was added in Figure 3 to illustrate the zero conditional covariance case. The item vector for item *x* lies along the vector for θ_{TT}. The conditional covariances between item *x* and items 1 to 6 are zero.

The magnitude of item pair conditional covariances is related to the closeness of the item pair vectors' directions, the closeness of one of the item vectors to vector θ_{TT}, and the magnitude of the items' multidimensional discrimination parameters. The conditional covariance of an item pair increases as the angle between the item pair vectors decreases and as the angle between either of the item vectors and the vector θ_{TT} increases. The conditional covariance of an item pair is also positively correlated with the

item's multidimensional discrimination parameters. The characteristics of the item pair conditional covariances laid the foundation for using global methods based on conditional covariances to assess the dimensionality of a test. The three global methods to be discussed here, DIMTEST, HCA/CCPROX, and DETECT, utilize these characteristics in different ways to explore the dimensional structure of a test.

DIMTEST. DIMTEST (Froelich & Habing, 2001; Nandakumar & Stout, 1993; Stout, 1987) tests the tenability of the assumption of essential unidimensionality. A test is, first, partitioned into two subtests, the assessment subtest (AT) and the partitioning subtest (PT). The subtest items are selected in a way that, when a test is multidimensional, the AT subtest items represent one homogeneous dimension and the PT subtest items represent the composition of multiple dimensions. DIMTEST then tests the null hypothesis of essential unidimensionality by evaluating whether the sum of the conditional covariances of all item pairs in the AT is significantly greater than zero after conditioning on the PT subtest score. If a test is unidimensional, then the AT and PT items are measuring the same dimension and the sum of conditional covariances should be close to zero according to the essential unidimensionality assumption which states that a test of length N is essentially unidimensional if for all item pairs, i *and* j, and all θ,

$$\frac{1}{N(N-1)}\sum_{1\le i\ne j\le N}\left|Cov(U_i,U_j \mid \theta)\right| \approx 0.$$

If the AT subtest items represent a distinct dimension, then there will be local item dependence after conditioning on PT and the sum of conditional covariances should be significantly greater than zero. This procedure, as with most factor analytic methods, can be used in both exploratory and confirmatory modes. When no substantive support for the AT candidate items exists, the exploratory mode is adopted. Exploratory factor analysis is

used to set a fixed number of items (specified as a percentage of the total number of items) with the highest loadings on the second extracted factor from an unrotated linear factor solution as the AT items. When substantive support exists for the identification of a distinct cluster of items, the confirmatory mode is adopted where the AT subtest items can be specified. However, DIMTEST only tests whether a test is unidimensional or multidimensional. It does not give an estimate of the number of dimensions and the partition of items when a test is proven to be multidimensional.

HCA/CCPROX. HCA/CCPROX (Roussos, 1995; Roussos, Stout, & Marden, 1998) uses a proximity measure based on conditional covariances to conduct a hierarchical cluster analysis. The proximity measure determines how similar two item clusters are. It is calculated as a weighted sum of conditional covariances between items in two clusters. Different weighting schemes can be used such as the unweighted pair group method of averages (UPGMA; Sokal & Michener, 1958). The program starts with each item representing one cluster, progressively combines two clusters with the highest proximity, and ends with all items clustered together. For a test of N items, HCA/CCPROX will produce N solutions, with from one to N items in a cluster. The N solutions are all candidates for the best partitioning of items into different dimensions. The procedure is useful as an initial attempt for exploring the dimensional structure of a test, but its use must be supplemented by other procedures, such as DIMTEST (in the confirmatory mode) and DETECT, to select the best clustering solution.

DETECT. DETECT (Kim, 1994; Zhang & Stout, 1999b) uses a genetic algorithm to find, in various partitions of the test items, the one that maximizes the DETECT index, which is defined as the mean of the signed conditional covariances of all item pairs. The

resulting maximum DETECT index represents the amount of multidimensionality present in a test. The partition of items associated with the maximum DETECT index is provided to shed light on the dimensional structure of the test. An index of whether the partition of test items represents simple or complex structure is also provided. The program can be run in two modes, exploratory and cross-validated. For the exploratory mode, the genetic algorithm is used to identify the partition of items in a dataset that maximizes the DETECT index. For the cross-validated mode, two datasets are involved. The genetic algorithm is used to identify the partition of items for the first dataset, and then this partition is used for the second dataset to obtain a set of cross-validated indices. Details about DETECT will be fully described in the next chapter, which provides a review of the development and evaluation of DETECT.

DIMTEST and HCA/CCPROX are not suited for determining the dimensional structure of a multidimensional test. DETECT, on the other hand, is a better candidate for this purpose. It tests for the strength of multidimensionality, estimates the number of dimensions underlying a test, and identifies the primary dimension measured by each item. However, as pointed by Tate (2002), the ability of DETECT to uncover the dimensional structure underlying a test of complex structure is still not clear since the procedure identifies mutually exclusive clusters of items and is most useful when approximate simple structure prevails. The current study will focus on the DETECT procedure and try to answer the question of how DETECT performs when test data possess complex dimensional structure.

Chapter 3: Review of DETECT

DETECT was originally developed by Kim (1994). The DETECT procedure is based on the structure and properties of conditional covariances which are informative about the dimensional structure of a test (Zhang & Stout, 1999a). Since its proposal, at least 14 studies have been conducted using simulated and/or real data to refine estimation procedures used in DETECT, to investigate the performance of DETECT under different conditions, and to compare DETECT with other dimensionality assessment procedures (e.g., Gierl et al., in press; Nandakumar & Ackerman, 2004; Stout et al., 1996; Zhang, Yu, & Nandakumar, 2003; Zhang & Stout, 1999b). These studies provide a better understanding of the theoretical underpinning and statistical properties of DETECT. This chapter is organized in three sections. In the first section, the theoretical development of DETECT is introduced by reviewing studies on the structure and properties of conditional covariances, which led to the proposal of DETECT, and studies that proposed and refined the DETECT procedure. In the second section, studies evaluating the performance of DETECT and applying DETECT to real test data are reviewed and organized according to the data structure of the tests being analyzed. A summary of the literature is provided in the final section.

Theoretical Development of DETECT

Properties of Conditional Covariances

The idea of using conditional covariances to investigate the dimensional structure of a test grew out of Stout's (1987) conceptualization of "essential unidimensionality." A test is considered essentially unidimensional if the item pair conditional covariances are close to zero given a unidimensional latent composite. This means responses to items on

a test are essentially independent from each other with close to zero conditional covariances despite the influence of possible trivial dimensions. Represented mathematically, a test of length N is said to be essentially unidimensional if for all item pairs, i and j, and all θ,

$$\frac{1}{N(N-1)} \sum_{1 \le i \ne j \le N} \left| Cov(U_i, U_j \mid \theta) \right| \approx 0 .$$

The properties of item pair conditional covariances for unidimensional cases were investigated by Douglas, Kim, Habing, and Gao (1998), Holland and Rosenbaum (1986), and Junker (1993). These researchers showed that violation of the assumptions of unidimensionality, caused either by the existence of nuisance dimensions or by the existence of multiple traits, would result in conditional covariances other than zero for items measuring the nuisance dimensions or clusters of items measuring multiple dimensions. Douglas et al. (1998) described the sign behavior of conditional covariances under conditions of multidimensionality and provided a rationale using illustrative examples. However, mathematical proof or theoretical support was not provided.

Zhang and Stout (1999a) investigated, mathematically, the structure and properties of conditional covariances under the generalized compensatory multidimensional model, which laid the foundation for using conditional covariances to infer the dimensional structure of a test (see also Chapter 2 of Zhang, 1997). The conditional covariances were mathematically proven to exhibit a consistent sign behavior and to correlate with several factors. The conditional covariance for an item pair, i and j, is positive if items i and j measure similar ability dimensions (i.e., lie on the same side of the direction of best measurement of the composite test), negative if items i and j measure different ability dimensions (i.e., lie on the opposite side of the direction of best

measurement of the composite test), and zero if one of the items measures the composite ability θ_{TT} (i.e., lies along the direction of best measurement of the composite test). The sign behavior of conditional covariances makes it possible to investigate the dimensional structure of a test by identifying clusters of items that have positive within-cluster (the item pair comes from the same cluster) conditional covariances and negative between-cluster (the item pair comes from different clusters) conditional covariances. DETECT utilizes this mechanism to search for the partition of items into different dimensions that maximizes the sum of the signed conditional covariances to determine the dimensional structure of a test.

The magnitude of the conditional covariances is correlated negatively with the magnitude of the angle between the two item vectors for an item pair in the multidimensional space. Conversely, the magnitude of the conditional covariances is positively correlated with the discrimination parameter of the items.

Zhang and Stout (1999a) also discussed the estimation of the expected conditional covariances. Two basic types of conditional scores (used as the surrogate to θ_{TT}) can be used to estimate conditional covariances. The first one uses the total scores on the remaining items other than the two items in consideration as the conditional score while the second one uses the total scores as the conditional score. The former is known to have a positive bias, while the latter is known to have a negative bias (Holland & Rosenbaum, 1986; Junker, 1993). In order to reduce estimation bias, Zhang and Stout (1999a) used the equally weighted average of two estimates. The positively biased estimator of $E[Cov(X_i, X_j \mid \theta_{TT}]$ is calculated as:

$$\hat{E}_{+}[Cov(X_i, X_j \mid \theta_{TT}] = \sum_{k=0}^{N-2} \frac{J_k}{J} \widehat{Cov}(X_i, X_j \mid S = k),$$

where S equals the total score on the remaining items, excluding items i and j, J is the total number of examinees, J_k is the number of examinees obtaining a score of k on the remaining items, and $\widehat{Cov}(X_i, X_j \mid S = k)$ is the observed covariance between the scores on items i and j for examinees obtaining a score of k on the remaining items. The positive bias associated with this estimator was first documented by Holland and Rosenbaum (1986). The degree of bias decreases as the reliability of the test increases. Thus, the positive bias will be smaller as the test becomes longer.

The negatively biased estimator of $E[Cov(X_i, X_j \mid \theta_{TT}]$ is calculated as:

$$\hat{E}_{-}[Cov(X_i, X_j \mid \theta_{TT}] = \sum_{k=0}^{N} \frac{J_k}{J} \widehat{Cov}(X_i, X_j \mid T = k),$$

where T equals the total test score including items i and j, J is the total number of examinees, J_k is the number of examinees obtaining a total test score of k, and $\widehat{Cov}(X_i, X_j \mid T = k)$ is the observed covariance between the scores on items i and j for examinees obtaining a total test score of k. The negative bias of this estimator is caused by including the scores on the two items for which the conditional covariance is being calculated (Junker, 1993; Zhang & Stout, 1999a). Like the positive bias, the negative bias also decreases as the length of the test increases.

Zhang and Stout (1999a) showed that for a 40-item unidimensional test the negative and positive biases were very close to each other. Thus, the final DETECT conditional covariance estimator is calculated as the equally weighted average of the two estimates,

$$\hat{E}[Cov(X_i, X_j \mid \theta_{TT}] = \frac{\hat{E}_+[Cov(X_i, X_j \mid \theta_{TT}] + \hat{E}_-[Cov(X_i, X_j \mid \theta_{TT}]}{2}.$$

Proposal and Refinement of DETECT

Kim (1994) first proposed the DETECT procedure for determining the number of dimensions underlying a test, estimating the strength of multidimensionality, and identifying the items contributing to each dominant dimension. To estimate the strength of multidimensionality, Kim proposed the DETECT index, $\hat{D}(P)$:

$$\hat{D}(P) = \frac{2}{N(N-1)} \sum_{1 \le i_1 < i_2 \le N} \delta_{i_1 i_2}(P)[\widehat{Cov}_{i_1 i_2}(S) - \overline{Cov(S)}],$$

where N is the number of dichotomous items on a test, P denotes the partition of N items into k clusters, i_1 and i_2 are the two items in each pair, $\widehat{Cov}_{i_1 i_2}(S) - \overline{Cov(S)}$ is the bias corrected estimate of conditional covariance between items i_1 and i_2, and $\delta_{i_1 i_2}(P)$ is analogous to the Kronecker delta, which takes two values—

$$\delta_{i_1 i_2}(P) = \begin{cases} 1 & \textit{if items } i_1 \textit{ and } i_2 \textit{ are in the same cluster of } P, \\ -1 & \textit{otherwise.} \end{cases}$$

The partition, P^*, that maximizes $\hat{D}(P)$ is the dimensional structure of the test, and $\hat{D}(P^*)$ is called the maximum DETECT index (D_{max}). To identify the partition of items that maximizes D_{max}, Kim used the Hierarchical Agglomerative Cluster (HAC) analysis algorithm programmed by Roussos (1993). However, the $\hat{D}(P^*)$ values obtained from HAC do not always represent the maximal values since the HAC algorithm only considers up to N possible clustering of items with a test of N items. A centering technique was used to correct the positive bias caused by using the S score as the

conditioning score. As shown in the formula for $\hat{D}(P)$, this is done by subtracting the mean of the estimates, $\overline{Cov}_{i_1 i_2}(S)$, from the conditional covariance estimates, $\widehat{Cov}_{i_1 i_2}(S)$.

According to Kim (1994), a D_{max} value of 0.1 or less indicates essential unidimensionality; a D_{max} value between 0.1 and 0.5 indicates weak multidimensionality; a D_{max} value between 0.5 and 1 indicates moderate multidimensionality; and a D_{max} value greater than 1 indicates strong multidimensionality. Although results from real and simulated data analyses were promising, the bias correction procedure still could be improved.

Since DETECT identifies mutually exclusive clusters of items, it works best when data display simple or approximate simple structure. To determine whether the partitioning, P^*, represents simple or complex structure, Kim (1994) proposed an index named, r_{max}, computed as the following ratio:

$$r_{max} = \frac{\hat{D}(P^*)}{\tilde{D}(P^*)}$$

where,

$$\tilde{D}(P) = \frac{2}{N(N-1)} \sum_{1 \le i_1 < i_2 \le N} \delta_{i_1 i_2}(P) \left| \widehat{Cov}_{i_1 i_2}(S) - \overline{Cov(S)} \right|,$$

and $\left| \widehat{Cov}_{i_1 i_2}(S) - \overline{Cov(S)} \right|$ is the absolute value of the bias corrected estimate of the conditional covariance between items i_1 and i_2. It is the maximum possible value that can be obtained by summing across all the corrected estimates of conditional covariances regardless of sign. If the partition, P^*, returns a strictly simple structure solution, then $\hat{D}(P^*)$ will be equal to $\tilde{D}(P^*)$, and the r_{max} will be one. If the partition, P^*, departs from

a strictly simple structure solution, then $\hat{D}(P^*)$ will be less than $\tilde{D}(P^*)$. This is because some of the bias corrected between-cluster conditional covariances could be positive due to the complexity of the data structure (besides the different dominant dimensions, two items in different clusters could measure another common dimension). When the negative sign of $\delta_{i_1 i_2}(P)$ is applied to these bias corrected between-cluster conditional covariances, the value of $\hat{D}(P^*)$ becomes less than the value of $\tilde{D}(P^*)$. The r_{max} will, then, be less than one. Values of r_{max} greater than or equal to 0.80 suggest the data display approximate simple structure, whereas values less than 0.80 suggest the data display complex structure (Kim, 1994).

Zhang and Stout (1999b), in the first part of their study, refined the DETECT procedure by providing a theoretical justification for DETECT, proposing a genetic algorithm to search for the partition that maximizes the DETECT index, and using a new bias correction procedure for estimating conditional covariances (see also Chapter 3 in Zhang, 1997). Their theoretical justification for DETECT was provided by defining the theoretical DETECT index and mathematically describing its behavior. The theoretical DETECT index is defined using the following formula:

$$D(P) = \frac{2}{N(N-1)} \sum_{1 \le i < j \le N} \delta_{ij} E[Cov(X_i, X_j \mid \theta_{TT})]$$

where $D(P)$ is called the theoretical DETECT index, N is the number of dichotomous items on a test, P denotes the partition of N items into k clusters, θ_{TT} is the test composite ability, X_i and X_j are observed scores on items i and j, $E[Cov(X_i, X_j \mid \theta_{TT}]$ is the expected conditional covariance between items i and j, and δ_{ij} is defined in the same way

as $\delta_{44}(P)$. For a test that is unidimensional, random clusters of items will be identified by DETECT. As a result, the within-cluster conditional covariances will be positive for some item pairs and negative for others, resulting in a value close to zero if summed up across all item pairs. However, for a test that is multidimensional, if DETECT successfully identifies the partition of items for which all the within-cluster conditional covariances are positive and all the between-cluster conditional covariances are negative, then the sum of the signed conditional covariances will be positive and equal to the maximum possible value. The theoretical DETECT index of $D(P)$ is operationalized by using number correct score as a surrogate for composite ability, θ_{TT}, and using the average of $\widehat{Cov}(X_i, X_j \mid T = k)$ and $\widehat{Cov}(X_i, X_j \mid S = k)$ to estimate $E[Cov(X_i, X_j \mid \theta_{TT}]$ (Zhang & Stout, 1999a).

Genetic algorithms, which are often used for optimization problems, borrow ideas from genetics and evolution to mutate and generate offspring solutions from existing outcomes to search for the optimal solution (Zhang, 1997). Since computing $D(P)$ for all possible partitions of items in a test would be computationally inefficient, Zhang and Stout (1999b) proposed using HCA/CCPROX (Roussos, 1995) to generate an initial set of partitions as a starting point to run the genetic algorithm. This process is described in Zhang and Stout (1999b, see also Stout et al., 1996).

Zhang and Stout (1999b) and Kim (1994) used two different bias correction methods for calculating conditional covariances. In order to evaluate the adequacy of different bias correction methods for calculating conditional covariances, Zhang, Yu, and Nandakumar (2003) investigated four bias correction methods in a simulation study. These correction methods included: 1) conditional covariances estimated by conditioning

on the total scores of the remaining items and centered ($\widehat{Cov}_{i_1i_2}(S)-\overline{Cov(S)}$); 2) conditional covariances estimated as the average of two estimators, one conditioning T and the other conditioning on S ($\frac{\widehat{Cov}_{i_1i_2}(T)+\widehat{Cov}_{i_1i_2}(S)}{2}$); 3) conditional covariances estimated by conditioning on the total scores and centered ($\widehat{Cov}_{i_1i_2}(T)-\overline{Cov(T)}$); and 4) conditional covariances estimated as the average of the two centered estimators, $\widehat{Cov}_{i_1i_2}(S)-\overline{Cov(S)}$ and $\widehat{Cov}_{i_1i_2}(T)-\overline{Cov(T)}$. Six independent variables were studied, including method for estimating conditional covariances (four bias corrected estimates plus two original biased estimates, $\hat{E}_+$ and $\hat{E}_-$), number of dimensions (1, 2), sample size (500, 1000), test length (30, 60), angle between clusters of items for the two-dimensional case (90°, 70°, 50°, 30°, 10°, 0°), and item distribution for the two-dimensional case (items equally distributed into clusters, items unequally distributed into clusters). The dependent variables included classification accuracy computed as the percentage of items correctly classified into dimensions, and the D_{max} and r_{max} indices. Item parameters were generated from the estimated item parameters from the 1992 National Assessment of Educational Progress (NAEP) test data. One hundred replications were done for each condition in the study.

Results from the simulation indicated that performance of the two biased estimates of conditional covariances was very unstable and classification accuracy dropped 30% to 50% when the angle between dimensions decreased. Consequently, these two estimation methods were not compared to the other bias corrected estimation methods. The last four independent variables had an impact on the dependent variables. For the unidimensional case, sample size and test length had a negative impact on the

D_{max} index. For the two-dimensional case, angle between clusters of items had a positive impact on all three dependent variables. As the sample size increased, classification accuracy and the r_{max} index increased, while the D_{max} index decreased. As test length increased, the r_{max} index decreased. When items were unequally distributed into clusters, all three dependent variables were slightly lower than those obtained for datasets with items equally distributed into clusters. This indicated that DETECT had difficulty in identifying the dimensional structure of a test when an unequal number of items was present in different dimensions. When the different estimation methods were compared, the first bias corrected estimation method using $\widehat{Cov}_{i,i_2}(S) - \overline{Cov(S)}$ produced slightly better results for the D_{max} and r_{max} indices. The third bias correction method using $\widehat{Cov}_{i,i_2}(T) - \overline{Cov(T)}$ produced slightly better results for classification accuracy. However, the differences found were very small and not statistically tested for significance, which made it difficult to conclude which estimation method worked the best. As discussed in the previous section, Zhang and Stout (1999a) provided both mathematical justification as well as empirical evidence that the average without centering bias correction method worked reasonably well. The bias correction method produced estimated conditional covariances with much reduced bias than the two biased estimates. Thus, the DETECT procedure uses this approach to estimate the conditional covariances.

Evaluation and Application of DETECT

Since the proposal of DETECT, 12 studies have been conducted to evaluate different aspects of DETECT—its estimation bias, its performance by itself and relative to other procedures—and to use DETECT with real as opposed to simulated tests (e.g.,

Finch & Habing, 2005; Roussos & Ozbek, 2003; Zhang & Stout, 1999b). These studies were conducted with simulated and/or real test data that had different structures. Data can assume simple, approximate simple or complex structures. As the review of these evaluation and application studies will reveal, diverse results were obtained.

Studies Using Data that Focus on Simple or Approximate Simple Dimensional Structures

Monahan, Stump, Finch, and Hambleton (2005) and Roussos and Ozbek (2003) evaluated bias associated with the estimators of the DETECT index (D_{max}) and the conditional covariances through simulation. Monahan et al. (2005) considered only the bias of the estimated D_{max} index for the unidimensional case. Four independent variables were studied: test length (5, 10, 15, 20, 40, 80), sample size (100, 500, 1000, 5000), IRT model used to generate data (1PL, 2PL, 3PL), and type of calculation method for the D_{max} index (exploratory, cross-validated). The dependent variable was the bias associated with the D_{max} index. The bias was operationalized as the departure of mean D_{max} values across replications from zero since the D_{max} index should be zero for the unidimensional case. The item parameters used for simulation were adopted from a state mathematics exam in the United States. According to the assumption of essential unidimensionality and the sign behavior of conditional covariances, the D_{max} index should be zero for unidimensional tests. Results from the study showed that under the null hypothesis of unidimensionality ($D_{max} = 0$), the cross-validated estimates of D_{max} index had a better control over bias ($D_{max} = 0.21$, the bias was 0.21) than the exploratory estimates of D_{max} index ($D_{max} = 0.39$, the bias was 0.39) across all studied conditions. Because the focus of

the present study is on multidimensional data of different structures, the Monahan et al. (2005) study is not reviewed and discussed in detail.

Roussos and Ozbek (2003) investigated both the unidimensional and multidimensional cases and evaluated not only the bias of the D_{max} and conditional covariance estimates but also the accuracy of classification when multidimensional data were simulated. Only data of simple or approximate simple structure were simulated. The independent variables included the number of dimensions (1, 2, 3), test length (5, 10, 20, 40), correlation between dimensions (0.50, 0.70), item distribution into clusters (equal and unequal), and type of item dimensionality structure (simple, approximate simple with item vectors in a fan of 15°, approximate simple with item vectors in a fan of 30°). The dependent variables included D_{max} bias, accuracy in clusters, IDN index (percentage of item pairs for which the sign of the conditional covariances was correctly estimated), average conditional covariance bias, and root mean square (RMS) conditional covariance bias. Item parameters were set in ranges that were typical in standardized tests (0.5 to 2.0 for item discrimination parameters, and -1.5 to 1.5 for item difficulty parameters). These parameters, as admitted by the authors, did not correspond perfectly to any real dataset, which limited the generalizability of the study. Although this is a simulation study, the authors did not mention the number of replications done for each condition.

Results from the study showed that DETECT had adequate control over bias (the D_{max} biases were 0.07 or less in all studied conditions for the multidimensional cases) for tests with 20 or more items. This is because the biases associated with the conditional covariance estimates were relatively small and the biases tended to cancel each other out for the within- and between-cluster estimates. While the correlations between dimensions,

dimensional structure, and item distribution did not show any influence on the bias associated with D_{max} and conditional covariance, they did influence the values of D_{max}. Higher correlations between dimensions, larger departures from simple structure, and unequal item distribution were associated with lower values of D_{max}. Classification accuracy results were all higher than 90% for the simulated multidimensional conditions indicating the adequacy of DETECT under simple or approximate simple structure conditions. This study showed that DETECT, with adequate control over bias and fairly high classification accuracy, was suitable for analyzing the dimensional structure of a test if the data displayed simple or approximate simple structure. However, it should be noted that the discrimination parameters used in the simulation were relatively high (i.e., 0.5 to 2.0), and since higher discrimination parameters are associated with higher estimates of conditional covariance, which, in turn, produce higher D_{max} values, high classification accuracy obtained in this study could be attributed to the item parameters used for the simulation.

Zhang and Stout (1999b), in the second part of their study, evaluated the performance of DETECT with simulated data as well as data from the Analytical Reasoning section of the Graduate Record Examination (GRE) and the Reading Comprehension section of the December 1991 Law School Admission Test (LSAT). Only data with approximate simple structure were simulated. Independent variables included the number of dimensions (1, 2, 3, 4), test length (20, 40), sample size (400, 800), and the presence of guessing in the unidimensional case. The dependent variables included the D_{max} and r_{max} indices and classification accuracy. Item parameters were

chosen to be representative of those from the LSAT. One hundred replications were done for each condition.

Simulation results showed that DETECT adequately estimated the D_{max} and r_{max} indices (D_{max} was less than 0.1 in the unidimensional cases and higher than 0.85 in the multidimensional cases; r_{max} was higher than 0.9 for all multidimensional cases). The number of runs with correct classification was greater than 90% for all the simulated conditions. The favorable results found in the simulation are limited by the influence of high discrimination parameters used in the simulation (i.e., 0.5 to 2.0). Two of the independent variables studied, test length and sample size, were shown to have a positive influence, while the number of dimensions was shown to have a negative influence on the dependent variables. In the analysis of the real test data, DETECT worked reasonably well identifying both sections as multidimensional and uncovering successfully the passage-based dimensions.

Only three studies in which the performance of DETECT was evaluated relative to other dimensionality assessment procedures were found in the literature (van Abswoude, van der Ark, & Sijtsma, 2004; Finch & Habing, 2005; Mroch & Bolt, 2006). The results of these studies revealed both strengths and weaknesses with DETECT. Van Abswoude et al. (2004) compared four nonparametric procedures: Mokken Scale Analysis for Polytomous Items (MSP, Molenaar & Sijtsma, 2000), DETECT, HCA/CCPROX, and DIMTEST. MSP is a nonparametric dimensionality assessment procedure based on normalized unconditional covariances. Van Abswoude et al. (2004) considered only data of simple structure. The independent variables included the dimensionality assessment procedure used (MSP, DETECT, HCA/CCPROX, DIMTEST),

IRT model used for simulating the data (2PL model, five-parameter acceleration model [5-PAM, Sijtsma & van der Ark, 2001]), number of dimensions (2, 4), correlation between dimensions (0.0, 0.2, 0.4, 0.6, 0.8, 1), test length (14, 28, 42, 56, 84), and item discrimination parameter (high, low). The dependent variables included the classification accuracy and the adequacy of the D_{max} index for DETECT and the T statistic for DIMTEST. Item parameters used for simulation were claimed to resemble parameter estimates from real test data (item discrimination parameters distributed with mean of 0.75 and standard deviation of 0.1, and item difficulty parameters set in the range from -2.0 to 2.0). However, it was not specified which test or which type of test the item parameters were generated from. While five replications were done for 27 studied conditions to check for stability of results from different procedures, only one dataset was simulated for the other studied conditions.

It was shown that the conditional covariance based procedures (DETECT and HCA/CCPROX) were superior to MSP in identifying the dimensional structure of the simulated tests (higher classification accuracy for studied conditions). The performance of all three procedures dropped as the correlations between dimensions increased. In general, DETECT performed better than HCA/CCPROX in uncovering the dimensional structure except for situations where the discrimination parameters of items were low and tests were long. The adequacy of the D_{max} index for DETECT was evaluated relative to the T statistic for DIMTEST. Both statistics were negatively influenced by the correlation between dimensions and positively influenced by the discrimination parameters of items. However, the D_{max} index was negatively influenced when unequal numbers of items

were distributed into clusters, while the *T* statistic was negatively influenced when equal numbers of items were distributed into clusters.

Based on the results obtained from the study, van Abswoude, et al. (2004) recommended using all of the procedures in data analysis and then to select the best one. However, it should be noted that the different procedures function in different ways. HCA/CCPROX produces a set of dimensional solutions, and one has to pick from among them the correct solution. DIMTEST tests for the presence of multidimensionality, but does not estimate the number of dimensions or identify the dimensional structure of a test. Using these procedures together could bring more light into the dimensional structure of a test, but it could also potentially complicate the situation by providing too much mixed information.

Mroch and Bolt (2006) compared three dimensionality assessment procedures. Two of them were nonparametric procedures: DETECT and MSP. The third procedure was a parametric procedure that grouped items based on their estimated discrimination parameters and was referred to as parametric cluster analysis (PCA; Miller & Hirsch, 1992). Only data of simple and approximate simple structures were simulated. The independent variables included sample size (250, 1000), number of dimensions (2, 3, 4), correlation between dimensions (matrix of equal correlations, matrix of unequal correlations of high and moderate values, matrix of unequal correlations of high, moderate, and low values), item distribution into clusters (equal, unequal), data structure (simple, approximate simple), and data generation model (MIRT 2PL compensatory model [M2PL], MIRT 2PL noncompensatory model [M2PLN]). The dependent variable, similarity coefficient (SM coefficient), was the percentage of item pairs accurately

matched according to their cluster membership. One hundred replications were conducted for each condition. A six-way ANOVA was conducted to evaluate the effects of the six independent variables. Paired-sample *t* tests and Cohen's *d* effect sizes were used to compare the three dimensionality assessment methods.

Results from the study suggested that DETECT and PCA performed quite similarly (the obtained SM coefficients were not significantly different from each other with $d = 0.02$) while both procedures outperformed MSP with significant differences between SM coefficients ($d > 1$). The correlation between dimensions affected DETECT and MSP more than PCA (larger effect size measures were obtained from ANOVA for DETECT and MSP). SM coefficients decreased for all three methods when correlations between dimensions increased. Data structure affected PCA the most (partial $\eta^2 = 0.59$), but DETECT and MSP were also substantially affected ($0.29 \leq$ partial $\eta^2 \leq 0.45$). Lower SM coefficients were obtained for the approximate simple condition than the simple condition for all three methods. The parametric procedure, PCA, was affected more by reduction in sample size than the two nonparametric procedures, DETECT and MSP. When an unequal number of items was put into each cluster, DETECT seemed to be affected the most, although the effect was small (partial $\eta^2 = 0.01$). Different data generation models did not show significant effect for all three procedures suggesting that PCA was robust to model misspecification. In summary, Mroch and Bolt (2006) found that DETECT and PCA were more preferable for dimensionality assessment than MSP.

Finch and Habing (2005, see also Finch, 2003) compared the performance of NOHARM and DETECT using both simulated and real data from a statewide reading instrument. Only data of approximate simple structure were simulated. The following

independent variables were considered: the number of dimensions (2, 6), skewness of the latent traits (-1.5, -0.5, 0, 0.5, 1.5), difficulty level of item parameters (average representing those of a basic skill exam, moderate representing those of SAT), presence of guessing, test length (15, 30, 60), sample size (1000, 2000), and correlations between dimensions (0, 0.30, 0.80, 0.95). The dependent variables included the number of dimensions identified and classification accuracy. Two sets of item parameters were used, one based on a statewide basic skill exam (mean of 0.97 and standard deviation of 0.32 for discrimination parameters, and mean of -0.92 and standard deviation of 0.76 for difficulty parameters) and the other based on the SAT (a lognormal distribution with mean of 0.00 and standard deviation of 0.35 for discrimination parameters, and mean of 0 and standard deviation of 1 for difficulty parameters). Five hundred replications were conducted for each condition.

Finch and Habing (2005) found that overall NOHARM and DETECT performed comparably in identifying the number of dimensions underlying a test and classifying items into correct clusters. However, under various conditions, the procedures performed differentially. DETECT was shown to perform better in classifying items into the correct clusters when the number of dimensions was low at two, while NOHARM was shown to perform better in identifying the number of dimensions. When error was made about the number of dimensions, DETECT tended to overestimate the number of dimensions for conditions with lower numbers of dimensions and underestimate the number of dimensions for conditions with higher numbers of dimensions. However, the number of dimensions estimated by NOHARM was generally close to the true number of dimensions simulated. Overall, the classification errors were lower for DETECT than for

NOHARM. When classification errors were made, DETECT tended to falsely separate items that should belong together, and NOHARM tended to combine items that should be separated. Number of dimensions and correlation between dimensions both showed a negative impact on the performance of the two procedures. Neither sample size nor test length showed a clear impact on the performance of the procedures. Both procedures performed better when no guessing was involved in the data. The real data analysis confirmed the results found in the simulation. Although NOHARM ($d = 5$) came closer than DETECT ($d = 3$) to identifying the expected number of dimensions ($d = 6$), DETECT grouped the items more consistently with the paragraphs in the exam and produced lower classification error rate. This study showed that DETECT performed inadequately when higher number of dimensions were involved (six in this case) even if data of approximate simple structure were simulated. This study also indicated that DETECT did not perform as well for tests simulated with typical difficulty than for tests simulated with low difficulty (classification accuracies were consistently lower for the tests simulated with the SAT parameters). However, since the magnitude of conditional covariances are positively influenced by the discrimination parameter of items, not the difficulty of the items, the lower classification accuracy results found in this study for the SAT parameters could be attributable to the lower discriminating parameters of the SAT than those of the basic skill exam.

Studies Using Data that Focus on Complex Dimensional Structures

Nandakumar and Ackerman (2004), in a book chapter, proposed an algorithm for combining DIMTEST and DETECT. Six steps were involved in the algorithm which first uses DIMTEST and DETECT sequentially and then iteratively to identify the minimum

number of clusters of items representing homogeneous unidimensional traits (DIMTEST is always ran before DETECT, and the algorithm stops whenever DIMTEST identifies a test or a subsection of the test as unidimensional). This study is different from the previously reviewed studies in that data of both approximate simple and complex structure were simulated to evaluate this new algorithm. Independent variables included the number of dimensions (1, 2), dimensional structure of multidimensional data (approximate simple, complex), and correlation between dimensions (0.50, 0.70). Dependent variables included the D_{max} index and classification accuracy. Item parameters were selected from estimated parameters from several nationally administered standardized achievement tests in the United States. Only one dataset was simulated for each studied condition.

The algorithm performed adequately for the unidimensional case and the simple structure conditions in the multidimensional cases. DIMTEST identified all the unidimensional tests as unidimensional eliminating the need to run DETECT. DIMTEST identified all the multidimensional tests as multidimensional in the approximate simple structure conditions, and DETECT identified the true dimensional structure in these conditions by correctly classifying all the items. However, for the complex structure conditions in the multidimensional cases, the algorithm performed less desirably for the 0.50 correlation condition and poorly for the 0.70 correlation condition. The clusters of items identified for the 0.50 correlation condition were still close to the true dimensional structure, but for the 0.70 correlation condition the algorithm stopped at the initial step since DIMTEST identified the test as unidimensional. This is the first study that assessed the performance of DETECT using simulated datasets that assumed complex data

structure. However, only one dataset was simulated for each correlation condition and DETECT was used only for one of the two datasets, thus limiting the inferences that could be drawn from the study regarding the performance of DETECT under conditions of complex structure.

Besides studies evaluating DETECT through simulation, researchers have also tried to analyze real test data with DETECT. Stout et al. (1996) used DIMTEST, HCA/CCPROX, and DETECT to investigate the dimensional structure of the December 1991, June 1992, and October 1992 administrations of the LSAT (see also Douglas, Kim, Roussos, Stout, & Zhang, 1999). Three subtests of the test were studied: the logical reasoning (LR) subtest, the analytical reasoning (AR) subtest, and the reading comprehension (RC) subtest. For the AR and RC subtests, the D_{max} and r_{max} values obtained indicated moderate to strong multidimensionality with simple structure. DETECT performed perfectly classifying items into passage-based clusters for these two subtests except for one analysis in which DETECT combined two science passages into one cluster for the RC subtest of the December 1991 administration. These results are reasonably close to the dimensional structure of these two subtests. However, for the LR subtest, the obtained D_{max} and r_{max} values indicated weak to no multidimensionality with complex structure. DETECT identified inconsistent clusters of items across administrations making it difficult to determine the dimensional structure of the LR subtest. Similar results were found in other studies applying DETECT to investigate the dimensional structures of test data from the School Achievement Indicators Program (SAIP, Leighton et al., in press), the National Assessment of Educational Progress (NAEP, Uribe-Zarain, Nandakumar, & Yu, 2005), and the SAT (Gierl et al., 2005).

The above four real data studies highlight DETECT's deficiency in analyzing data of complex structure. This situation is unfortunate because many real testing situations involve data that display complex structure (e.g., Nandakumar & Ackerman, 2004; Stout et al., 1996). However, after a review of the literature, only one article was found investigating the performance of DETECT systematically through simulation under the conditions of complex structure (Gierl et al., in press).

Gierl et al. (in press) investigated the performance of DETECT under conditions of both approximate simple and complex structure using simulated as well as real data from the SAT and the SAIP. The independent variables included in their study were the degree of complexity in data structure (0%, 10%, 30%, 50%), correlation between dimensions (0.00, 0.30, 0.60, 0.90), and sample size (500, 1000, 1500). The dependent variables included classification accuracy and consistency. The item parameters were selected to resemble those from the LSAT and set in the same range as those in Zhang and Stout's (1999b) study. One hundred replications were conducted for each condition.

Simulation results from the study suggested that DETECT worked well, using a criterion that 90% of the items be partitioned into the correct clusters, for 31 of 45 complex data structure conditions. Correlation between dimensions was found to have a noticeable impact on the performance of DETECT. For correlations of 0.60 or lower, DETECT worked above the criterion of 90% even for the three complex data structure conditions given adequate sample size (1000 for the complex 30% condition, and 1500 for the complex 50% condition). When the correlation was 0.75, DETECT worked above criterion only for the approximate simple, complex 10%, and complex 30% conditions given the sample size for the complex 30% conditions was 1000 or higher. Classification

rates for the complex 50% conditions dropped to around 80%. However, for the correlation of 0.90, DETECT worked poorly for all complex data structure conditions (classification rates dropped from above criterion to less than 50% for the complex 50% conditions). Correlation between dimensions and degree of complexity both influenced classification rates negatively. In contrast, sample size was found to influence classification accuracy positively (higher sample sizes produced higher classification rates).

The real data analyses were conducted with two datasets extracted from the SAT and the SAIP. A two-dimensional approximate simple structure was hypothesized for the SAT dataset containing two dimensions according to content areas, Math and Critical Reading, with moderate correlation between dimensions. The SAIP dataset, on the other hand, was hypothesized as having a two-dimensional complex structure containing two dimensions according to item types, multiple choice and constructed response items, with high correlation between dimensions. Results from the analyses of the two datasets had good correspondence with the simulation results. The SAT dataset was identified as multidimensional, and 96% of items were correctly and consistently classified into two clusters. In contrast, the SAIP dataset were identified as weakly multidimensional and only 45% of items were correctly and consistently classified into two clusters.

Despite these outcomes, four questions still remain unanswered about the performance of DETECT with items that display complex structure. First, it is very likely that a test of complex structure will have more than 50% of its items measuring multiple dimensions. For example, when the math section of the SAT 2003 field trial data was analyzed by setting the number of dimensions to two based on exploratory factor analysis

results, a correlation of 0.69 between the dimensions was obtained using NOHARM (Gierl et al., 2005). When the data was analyzed with DETECT, a classification consistency of 44% was obtained across two samples. This result is much lower than the corresponding classification consistency result obtained in the Gierl et al. (in press) study (for the 0.75 correlation and complex 50% condition, the classification consistency was around 75%). This result suggests that the SAT math section could have a degree of complexity higher than 50%, which led to the low classification consistency. Furthermore, research on cognitive processes suggests that cognitive skills do not operate in isolation but function in a network of interrelated processes (e.g., Kuhn, 2001; Vosniadou & Brewer, 1992). This interrelatedness will likely cause tests to display higher degrees of complexity. Thus, whether DETECT will perform satisfactorily for correlations of 0.60 or lower when higher degrees of complexity are present needs to be investigated.

Second, for most educational and psychological tests where social science constructs are involved, the correlations between dimensions are moderate to high (e.g., Anastasi & Urbina, 1996; Sattler, 2001). Thus, the use of correlations such as 0.0 and 0.3 is, to some degree, unrealistic. As shown by the simulation results in the Gierl et al. (in press) study, DETECT classification accuracy and consistency dropped dramatically (from above 90% to below 50% in the complex 50% cases) when the correlation went from 0.60 to 0.90. It will be informative and meaningful to set finer intervals for investigation between the correlations of 0.60 and 0.90 since these correlations are most commonly expected on educational and psychological tests.

Third, as the magnitude of the conditional covariances is positively correlated with item discrimination, the impact of discrimination parameter of items on the

performance of DETECT should be investigated. The discrimination indices of the items used for simulation in the Gierl et al. (in press) study were moderate to high (0.5 to 1.1). Therefore, it is necessary to investigate the performance of DETECT for tests with a wider range of discrimination indices. Fourth, the effect of larger sample sizes (e.g., up to 2500) should be studied since larger sample sizes are commonly found in field tests of large-scale testing programs.

Summary

DETECT, as a dimensionality assessment procedure with a much shorter history than most factor analytic procedures, has been investigated in 14 studies using simulated and real data. The review of these studies presented in this chapter shows that DETECT performed quite well identifying the true number of dimensions and the correct item clusters associated with different dimensions when data possessed simple or approximate simple structure (e.g., Stout et al., 1996; Zhang & Stout, 1999b; Nandakumar & Ackerman, 2004). However, when complex structure was involved, DETECT performed inadequately and inconsistently across some study conditions (e.g., Gierl et al., 2005; Nandakumar & Ackerman, 2004; Stout et al., 1996). While the existence of approximate simple structure in test data is known for some large-scale tests (e.g., the AR and RC subtests of the LSAT), for many realistic testing situations the test data display a complex dimensional structure (Gierl et al., in press; Nandakumar & Ackerman, 2004). Since DETECT identifies mutually exclusive and dimensionally homogeneous clusters of items through analysis of the conditional covariance matrix, it works best for data of simple or approximate simple structures and might be problematic for analyzing data of complex

structure (Zhang & Stout, 1999b). However, the use of DETECT would still be meaningful since Zhang and Stout (1999b) claimed:

> It is very important to note that DETECT is still informative when approximate simple structure fails to hold. In particular, it can still locate relatively dimensionally homogeneous clusters; however, there is no longer a unique 'best' or 'correct' partition to be found by DETECT because there will be little to no separation between some of the clusters found. (p. 215)

Hence, studies evaluating the properties and performance of DETECT under conditions of complex data structure deserve more attention. However, only one published study was found studying this issue systematically through simulation (Gierl et al., in press). Results from the Gierl et al. (in press) study shed some light on the restrictions that need to be satisfied in order for DETECT to perform adequately for test data of complex structure. However, as discussed in the previous section, many questions were still left unanswered. The present study was thus proposed to answer these questions.

Chapter 4: Method

The study was completed in two stages. In stage one, simulation studies were conducted to evaluate the performance of DETECT when data displayed approximate simple and different degrees of complex structure. This evaluation was made in terms of the D_{max} and r_{max} indices and the accuracy of the classification results. The impact of three factors—degree of complexity in data structure, correlation between dimensions, and sample size—was studied. The relationship between the D_{max} index and classification accuracy and the relationship between the r_{max} index and classification accuracy were also studied. In stage two, real data studies were conducted in which DETECT was applied to the SAT 2005 March administration data to check for consistency between the results from the real and the simulated conditions. Results obtained from both stages were then used to develop new guidelines and recommendations for using and interpreting DETECT results under conditions of both approximate simple and complex data structure.

Stage 1: Simulation Studies

Data

Examinee responses to a 40-item test were simulated with two different sets of item parameters, one based on the LSAT and the other on the SAT. The LSAT represents a large-scale test with items having moderate to relatively high discrimination parameters (i.e., range from 0.5 to 1.1). The SAT, on the other hand, represents a large-scale test with items having a range of low to high discrimination parameters (i.e., range from 0.2 to 1.3). These two sets of item parameters were used to simulate different real testing situations where large-scale tests were involved. Since the discrimination parameters of

the items for the two tests overlapped, they cannot be used to determine the impact of item discrimination parameter on the performance of DETECT. However, the use of these two parameter sets could shed some light on the possible impact of item discrimination parameter if differences were to be found.

To keep the study sharp and focused, only two-dimensional data were simulated. An extension of the Gierl et al. (in press) study, the intent was to gain a fuller understanding the performance of DETECT with the existence of complex structures in the two-dimensional case. The length of the test was also fixed. Simulation studies suggested that DETECT estimation bias was well controlled for tests that had 20 or more items resulting in high classification accuracy (Roussos & Ozbek, 2003; Zhang & Stout, 1999b). Thus, 40 items were simulated for all studied conditions.

The data were simulated using the compensatory multidimensional item response theory (MIRT) model (Reckase, 1997). The 2PL item response function for the compensatory MIRT model can be expressed using the following formula:

$$P_i[U_i = 1 | (\theta_1, ..., \theta_k)] = \frac{1}{1 + e^{-1.7(a_{i1}\theta_1 + a_{i2}\theta_2 + ... a_{ik}\theta_k + d_i)}},$$

where U_i is the response to item i, $\vec{\theta}^T = (\theta_1, ..., \theta_k)$ is the examinee ability vector, $\vec{a}_i^T = (a_1, ..., a_k)$ is the item discrimination vector, d_i is the item difficulty parameter, and k is the number of dimensions underlying the test. The examinee abilities were assumed to have a bivariate normal distribution with a mean of (0, 0) and a standard deviation of (1, 1). There are reasons why the 2PL MIRT model was adopted. First, for tests such as the SAT, formula scoring is used where points are deducted for incorrect answers to multiple-choice items to penalize guessing. Being aware of the formula scoring

procedure, examinees tend to omit questions instead of eliminating one and randomly guessing among others (Oh & Reshetar, 2004). Thus, guessing becomes a less prominent issue in tests such as the SAT. Although the LSAT does not use formula scoring and is still prone to guessing, most researchers using the LSAT parameters for simulations adopted the 2PL model (Gierl et al., in press; Roussos & Ozbek, 2003; Zhang & Stout, 1999b). Second, studies have shown that dimensionality assessment procedures including DETECT and NOHARM perform better when guessing is not present in the data (Finch & Habing, 2005; Zhang & Stout, 1999b). Based on these two considerations, the presence of guessing was not studied, and the 2PL MIRT model was used for data simulation. The first set of item parameters, a_1-, a_2-, and d-parameters, was adopted from the Gierl et al. (in press) study to resemble multidimensional tests like the LSAT. The second set of item parameters was obtained from analysis of the SAT 2003 field trial data (Gierl et al., 2005).

LSAT item parameters. When data of approximate simple structure were simulated, for items measuring dimension one, the a_1-parameters were set in the range of 0.5 to 1.1 with an increment of 0.2, whereas the a_2-parameters were set in the range of 0.05 to 0.20 with an increment of 0.05. The values of the a_1- and a_2-parameters were set in the opposite way from those of the dimension one items for items measuring dimension two. The d-parameters for all items ranged from -1 to 1 with an increment of 0.5. The angular directions of the dimension one items ranged from 5.71° to 10.30°, and those of the dimension two items ranged from 79.66° to 84.26°. The angular directions were both within 20 degrees from the x- or y-axis and were representative of an approximate simple structure solution (Froelich & Habing, 2001). The item parameters

and the angular directions of the items are presented in Table 2. Figure 5 contains the vector plot of the items for the approximate simple structure condition.

Table 2

Item Parameters for the Simulated Approximate Simple Structure Items Using the LSAT Parameters

Simple	Complex 40%	Complex 80%	a_1	a_2	d	Direction
1	X	X	0.50	0.05	-1.00	5.71
2			0.70	0.10	-0.50	8.13
3		X	0.90	0.15	0.00	9.46
4			1.10	0.20	0.50	10.30
5			0.50	0.05	1.00	5.71
6	X	X	0.70	0.10	-1.00	8.13
7	X	X	0.90	0.15	-0.50	9.46
8		X	1.10	0.20	0.00	10.30
9	X	X	0.50	0.05	0.50	5.71
10		X	0.70	0.10	1.00	8.13
11			0.90	0.15	-1.00	9.46
12	X	X	1.10	0.20	-0.50	10.30
13		X	0.50	0.05	0.00	5.71
14	X	X	0.70	0.10	0.50	8.13
15	X	X	0.90	0.15	1.00	9.46
16		X	1.10	0.20	-1.00	10.30
17		X	0.50	0.05	-0.50	5.71
18		X	0.70	0.10	0.00	8.13
19		X	0.90	0.15	0.50	9.46
20	X	X	1.10	0.20	1.00	10.30
21	X	X	0.05	0.50	-1.00	84.26
22	X	X	0.10	0.70	-0.50	81.84
23		X	0.15	0.90	0.00	80.51
24		X	0.20	1.10	0.50	79.66
25	X	X	0.05	0.50	1.00	84.26
26		X	0.10	0.70	-1.00	81.84
27		X	0.15	0.90	-0.50	80.51
28	X	X	0.20	1.10	0.00	79.66
29		X	0.05	0.50	0.50	84.26
30		X	0.10	0.70	1.00	81.84
31			0.15	0.90	-1.00	80.51
32		X	0.20	1.10	-0.50	79.66
33		X	0.05	0.50	0.00	84.26
34			0.10	0.70	0.50	81.84
35	X	X	0.15	0.90	1.00	80.51

Table 2con't

Simple	Complex 40%	Complex 80%	a_1	a_2	d	Direction
37			0.05	0.50	-0.50	84.26
38	X	X	0.10	0.70	0.00	81.84
39	X	X	0.15	0.90	0.50	80.51
40			0.20	1.10	1.00	79.66
Simple	Mean		0.46	0.46	0.00	44.98
	(SD)		(0.38)	(0.38)	(0.72)	(37.09)
Complex	Mean		0.46	0.46	0.00	44.98
40%	(SD)		(0.38)	(0.38)	(0.68)	(37.41)
Complex	Mean		0.46	0.46	0.00	44.98
80%	(SD)		(0.40)	(0.40)	(0.85)	(39.15)

Note. X indicates the item was omitted.

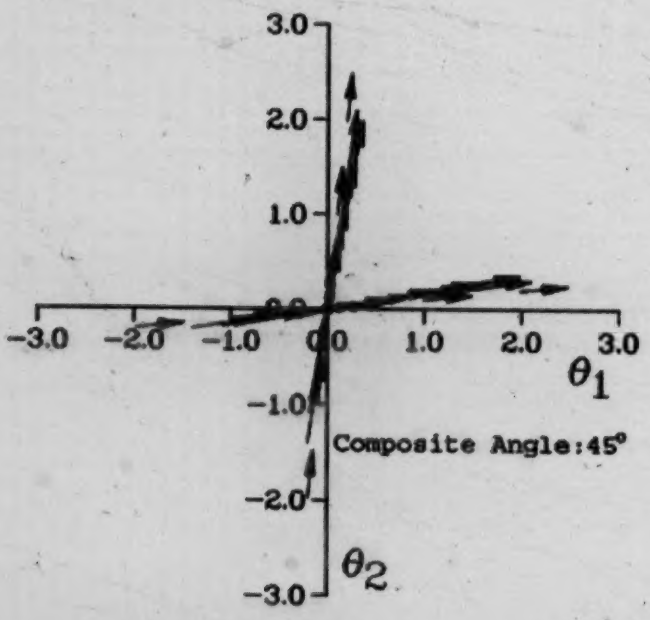


Figure 5. Vector plot of simulated items for the approximate simple structure condition using the LSAT parameters.

When data of complex structure were simulated, both the a_1- and a_2-parameters of the complex structure items were set within the range of 0.5 to 1.1 with an increment of 0.2, and the difference between the two was set to 0.2. The *d*-parameters remained the same. The angular directions of the complex structure items ranged from 35.52° to 54.44°,

which was within ten degrees from the test composite direction of 45°, representative of the complex structure. Two complex data structure conditions were included, complex 40% and complex 80%. For the complex 40% condition, eight items measuring dimension one and eight items measuring dimension two were replaced with 16 complex items measuring the composite of dimensions one and two. Figure 6 contains the vector plot of items in the complex 40% condition. For the complex 80% condition, 16 items

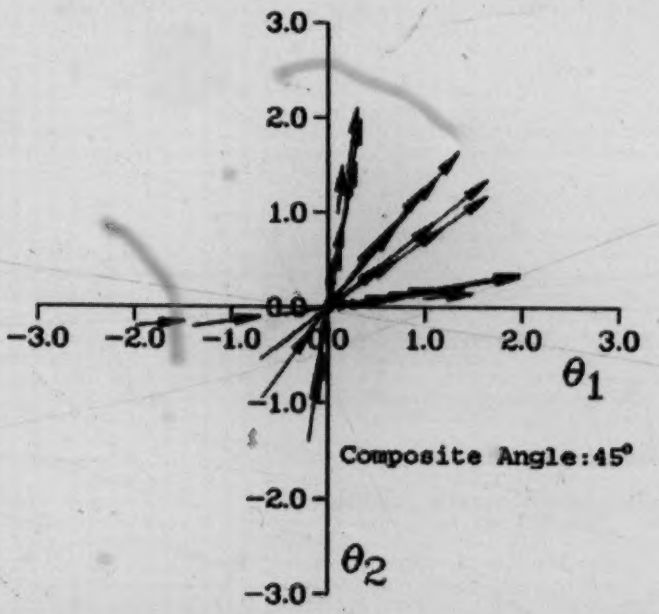


Figure 6. Vector plot of simulated items for the complex 40% structure condition using the LSAT parameters.

measuring dimension one and 16 items measuring dimension two were replaced with 32 complex items measuring the composite of dimensions one and two. Figure 7 contains the vector plot of items in the complex 80% condition.

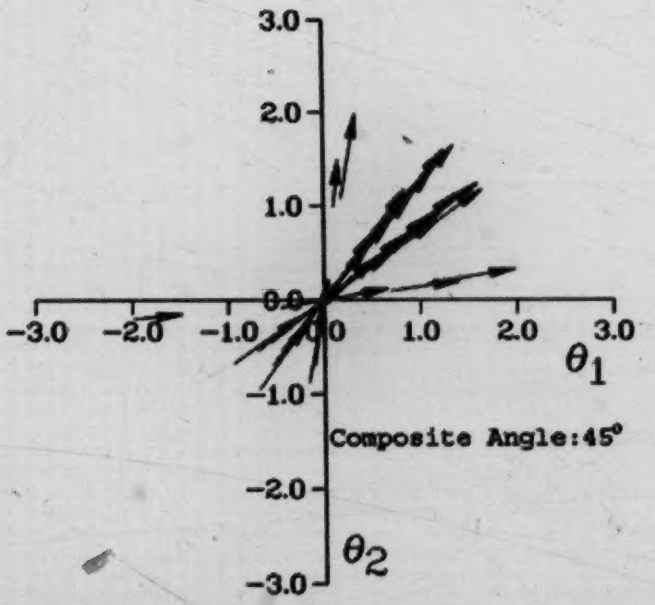


Figure 7. Vector plot of simulated items for the complex 80% structure condition using the LSAT parameters.

The item parameters and angular directions for all the three conditions are presented in Table 2 and Table 3. The item parameters were manipulated so that they were similar across conditions. The distributions (means and standard deviations) of item parameters across conditions were very similar as shown by the descriptive statistics in the bottom rows in Tables 2 and 3.

SAT item parameters. The estimated item parameters from the Math and Critical Reading subtests of the SAT 2003 field trial were used as the basis for simulation (cf., Gierl et al., 2005). The item parameters for the dimension one items were determined by the distribution of the Math items. The item parameters for the dimension two items were determined by the distribution of the Critical Reading items. When data of approximate simple structure were simulated, the a_1-parameters were set in the range of 0.40 to 1.30 with an increment of 0.10, whereas the a_2-parameters were set in the range of 0.00 to 0.18

Table 3

Item Parameters for the Simulated Complex Structure Items Using the LSAT Parameters

Simple	Complex 40%	Complex 80%	a_1	a_2	d	Direction
X			0.70	0.50	-1.00	35.52
X			0.70	0.50	-0.50	35.52
X			0.70	0.50	0.00	35.52
X	X		0.70	0.50	0.50	35.52
X	X		0.70	0.50	1.00	35.52
X	X		0.90	0.70	-1.00	37.86
X	X		0.90	0.70	-0.50	37.86
X	X		0.90	0.70	0.00	37.86
X			0.90	0.70	0.50	37.86
X			0.90	0.70	1.00	37.86
X	X		0.90	0.70	-1.00	37.86
X			1.10	0.90	-0.50	39.27
X			1.10	0.90	0.00	39.27
X			1.10	0.90	0.50	39.27
X	X		1.10	0.90	1.00	39.27
X	X		1.10	0.90	0.00	39.27
X			0.90	1.10	-1.00	50.69
X	X		0.90	1.10	-0.50	50.69
X			0.90	1.10	0.00	50.69
X	X		0.90	1.10	0.50	50.69
X			0.90	1.10	1.00	50.69
X			0.70	0.90	-1.00	52.10
X	X		0.70	0.90	-0.50	52.10
X	X		0.70	0.90	0.00	52.10
X			0.70	0.90	0.50	52.10
X	X		0.70	0.90	1.00	52.10
X	X		0.70	0.90	-1.00	52.10
X			0.50	0.70	-0.50	54.44
X			0.50	0.70	0.00	54.44
X	X		0.50	0.70	0.50	54.44
X			0.50	0.70	1.00	54.44
X	X		0.50	0.70	0.00	54.44
Simple	Mean (SD)		--	--	--	--
Complex 40%	Mean (SD)		0.80 (0.21)	0.80 (0.21)	0.00 (0.71)	44.98 (7.90)
Complex 80%	Mean (SD)		0.80 (0.19)	0.80 (0.19)	0.00 (0.70)	44.98 (7.68)

Note. X indicates the item was omitted.

with an increment of 0.02 for items measuring dimension one. For items measuring dimension two, the a_1-parameters were set in the range of 0.20 to 0.65 with an increment of 0.05, whereas the a_2-parameters were set in the range of 0.25 to 0.60 with an increment of 0.05. The *d*-parameters ranged from -1 to 1 with an increment of 0.5 for both cases. The angular directions of the dimension one items ranged from 0.00° to 7.88°, while the angular directions of the dimension two items ranged from 31.60° to 51.32°. Although the angular directions of the dimension two items were not within 20 degrees from the y-axis that corresponds to dimension two, there was a clear separation between the dimension one and dimension two items, indicating two independent item clusters. Analysis of the SAT 2003 field trial data showed that the r_{max} value for the two dimensional data was 0.86, indicating approximate simple structure. Thus, the definition of the approximate simple structure might be too limited by restricting the angular departure of items from its correspondent axes to 20 degrees. Analysis of the simulated data that approximated the SAT composite test was conducted to try to refine the definition of the approximate simple structure and the evaluation criterion for r_{max}. The item parameters and angular directions for the approximate simple structure items are presented in Table 4. Figure 8 contains the vector plot of the approximate simple structure condition.

Table 4

Item Parameters for the Simulated Approximate Simple Structure Items Using the SAT Parameters

Simple	Complex 40%	Complex 80%	a_1	a_2	d	Direction
1	X	X	1.30	0.18	-1.00	7.88
2		X	1.20	0.16	-0.50	7.59
3	X	X	1.10	0.14	0.00	7.25
4	X	X	1.00	0.12	0.50	6.84
5	X	X	0.90	0.10	1.00	6.34
6		X	0.80	0.08	-1.00	5.71
7		X	0.70	0.06	-0.50	4.90
8			0.60	0.04	0.00	3.81
9		X	0.50	0.02	0.50	2.29
10		X	0.40	0.00	1.00	0.00
11		X	1.30	0.18	1.00	7.88
12			1.20	0.16	0.50	7.59
13		X	1.10	0.14	0.00	7.25
14		X	1.00	0.12	-0.50	6.84
15		X	0.90	0.10	-1.00	6.34
16	X		0.80	0.08	1.00	5.71
17	X	X	0.70	0.06	0.50	4.90
18	X	X	0.60	0.04	0.00	3.81
19			0.50	0.02	-0.50	2.29
20	X	X	0.40	0.00	-1.00	0.00
21	X	X	0.65	0.40	-1.00	31.59
22		X	0.60	0.45	-0.50	36.86
23	X	X	0.55	0.50	0.00	42.26
24		X	0.50	0.55	0.50	47.71
25	X	X	0.45	0.60	1.00	53.11
26			0.40	0.45	-1.00	48.35
27	X	X	0.35	0.40	-0.50	48.79
28		X	0.30	0.35	0.00	49.38
29		X	0.25	0.30	0.50	50.17
30		X	0.20	0.25	1.00	51.32
31		X	0.65	0.40	1.00	31.59
32			0.60	0.45	0.50	36.86
33			0.55	0.50	0.00	42.26
34	X	X	0.50	0.55	-0.50	47.71
35		X	0.45	0.60	-1.00	53.11

Table 4con't

Simple	Complex 40%	Complex 80%	a_1	a_2	d	Direction
37		X	0.35	0.40	0.50	48.79
38	X	X	0.30	0.35	0.00	49.38
39			0.25	0.30	-0.50	50.17
40	X	X	0.20	0.25	-1.00	51.32
Simple	Mean		0.64	0.26	0.00	25.61
	(SD)		(0.31)	(0.19)	(0.72)	(21.20)
Complex	Mean		0.64	0.25	0.00	25.38
40%	(SD)		(0.32)	(0.19)	(0.69)	(21.23)
Complex	Mean		0.61	0.25	0.00	24.63
80%	(SD)		(0.29)	(0.20)	(0.65)	(21.57)

Note. X indicates the item was omitted.

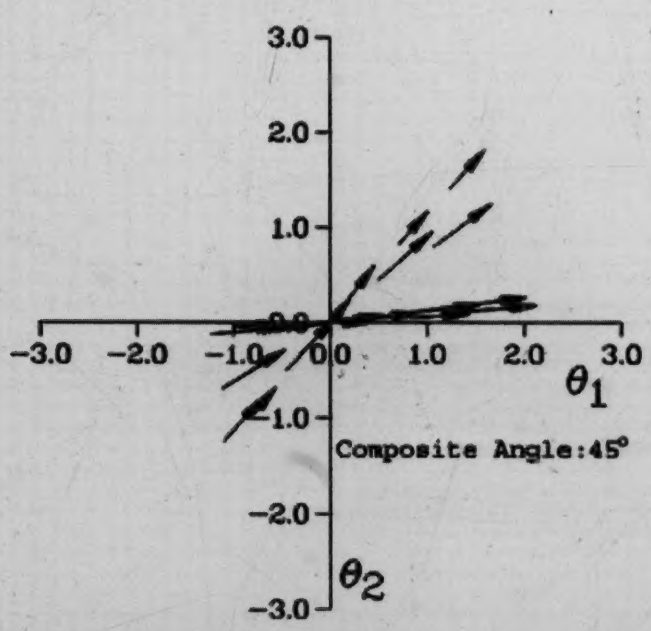


Figure 8. Vector plot of simulated items for the approximate simple structure condition using the SAT parameters.

When data of complex structure were simulated, the a_1-parameters of the first set of complex structure items (still measure more of dimension one) were set in the range of 0.50 to 1.10 with an increment of 0.20, whereas the a_2-parameters were set in the range of 0.17 to 0.32 with an increment of 0.05. The a_1-parameters of the second set of complex

structure items (still measure more of dimension two) were set in the range of 0.40 to 0.70 with an increment of 0.10, whereas the a_2-parameters were set in the range of 0.15 to 0.30 with an increment of 0.05. The *d*-parameters remained the same. The angular directions of the complex structure items ranged from 16.21° to 23.19°. The two complex structure conditions, complex 40% and complex 80%, were set in the same way as those for the data simulated using the LSAT item parameters. Figures 9 and 10 contain the vector plots for items in these two conditions. The item parameters and angular directions for all the three conditions are presented in Table 4 and Table 5. The descriptive statistics in the bottom rows indicated that the item parameters were similar (with similar means and standard deviations) across conditions.

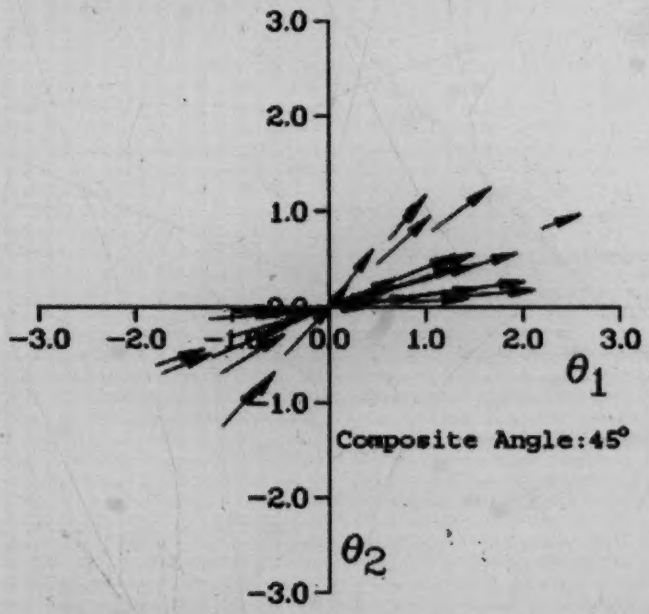


Figure 9. Vector plot of simulated items for the complex 40% structure condition using the SAT parameters.

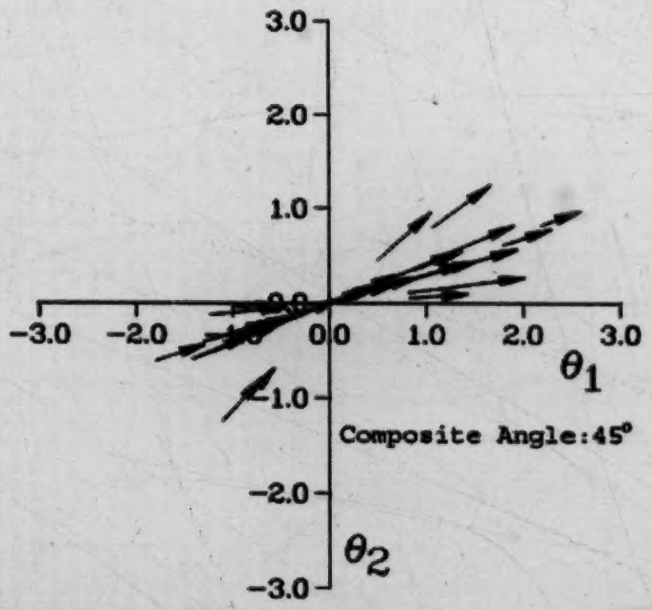


Figure 10. Vector plot of simulated items for the complex 80% structure condition using the SAT parameters.

Research Design

Three independent variables, including the degree of complex data structure, the correlation between dimensions, and the sample size, were manipulated in the current study to form a $3\times4\times3$ fully crossed design. The degree of complexity had three levels, 0%, 40%, and 80%. The complex 40% condition represented tests with lower degrees of complexity while the complex 80% condition represented tests with higher degrees of complexity. The correlation between dimensions had four levels, 0.6, 0.7, 0.8, and 0.9. The sample size had three levels, 1500, 2000, and 2500.

Table 5

Item Parameters for the Simulated Complex Structure Items Using the SAT Parameters

Simple	Complex 40%	Complex 80%	a_1	a_2	d	Direction
X			1.10	0.32	-1.00	16.21
X	X		0.90	0.27	-0.50	16.69
X			0.70	0.22	0.00	17.44
X	X		0.50	0.17	0.50	18.77
X	X		1.10	0.32	1.00	16.21
X			0.90	0.27	-1.00	16.69
X	X		0.70	0.22	-0.50	17.44
X			0.50	0.17	0.00	18.77
X			1.10	0.32	0.50	16.21
X			0.90	0.27	1.00	16.69
X	X		0.70	0.22	-1.00	17.44
X			0.50	0.17	-0.50	18.77
X	X		1.10	0.32	0.00	16.21
X	X		0.90	0.27	0.50	16.69
X			0.70	0.22	1.00	17.44
X	X		0.50	0.17	-1.00	18.77
X			0.40	0.15	-0.50	20.55
X			0.50	0.20	0.00	21.79
X	X		0.60	0.25	0.50	22.61
X	X		0.70	0.30	1.00	23.19
X			0.40	0.15	-1.00	20.55
X	X		0.50	0.20	-0.50	21.79
X			0.60	0.25	0.00	22.61
X			0.70	0.30	0.50	23.19
X	X		0.40	0.15	1.00	20.55
X	X		0.50	0.20	-1.00	21.79
X	X		0.60	0.25	-0.50	22.61
X	X		0.70	0.30	0.00	23.19
X	X		0.40	0.15	0.50	20.55
X			0.50	0.20	1.00	21.79
X			0.60	0.25	1.00	22.61
X			0.70	0.30	-1.00	23.19
Simple	Mean (SD)		--	--	--	--
Complex 40%	Mean (SD)		0.68 (0.22)	0.24 (0.06)	0.00 (0.77)	19.66 (2.66)
Complex 80%	Mean (SD)		0.68 (0.22)	0.24 (0.06)	0.00 (0.74)	19.66 (2.61)

Note. X indicates the item was omitted.

Data Analysis

The data were simulated using the MULTISIM software (The William Stout Institute for Measurement, 1993) and analyzed with DETECT. Each condition was replicated 100 times to obtain stable estimates of the indices and classification rates (Harwell, Stone, Hsu, & Kirisci, 1996). The Visual Basic code for batch processing the simulation and DETECT runs in EXCEL is included in Appendix A.

The four dependent variables included D_{max}, r_{max}, classification accuracy, and classification consistency. Classification accuracy was obtained by calculating the match between the DETECT item classification and the true item clustering as simulated. This match was calculated as the percentage of items partitioned into the correct dimensions. The mean and standard deviation of the match over 100 replications were used as the final statistics to evaluate classification accuracy. The Visual Basic code for batch processing DETECT outputs and calculating classification accuracy is included in Appendix B.

Classification consistency was used to evaluate cross-sample consistency of DETECT item classification. When analyzing real data, researchers rarely know the true underlying dimensional structure of the data. In most cases, they rely on cross-validation using results obtained from randomly equivalent samples to confirm the item classification produced by the testing sample. Thus, in the current study, two random samples were generated for each simulated condition, and the match between the classification results obtained from the two samples was used to evaluate DETECT classification consistency. The match was calculated as the percentage of items consistently partitioned into the same dimensions across the two samples. The mean and

standard deviation of this match over 100 replications were used as the final statistics for evaluating classification consistency. However, consistency does not always indicate accuracy. It is possible that a match between two samples on an item is actually wrong and the item belongs to a different dimension. In order to indicate what percentage of the matching rate might be due to error in different simulated conditions, an index called misclassification error was calculated as the difference between the matching rate and the correctly classified matching rate. By correctly classified matching rate, it means DETECT not only consistently classified an item into the same dimension across samples but also classified the item into the correct dimension it was simulated to measure. Since the truth is known in the simulation, the misclassification error can be obtained which indicates the rate of consistent misclassification. It helps researchers estimate the misclassification error rate under different conditions and determine whether the consistency found indicates the real dimensional structure. The mean and standard deviation of this index over 100 replications were used to evaluate misclassification error. Appendix C includes the Visual Basic code for calculating classification consistency.

The relationship between the D_{max} values and the classification accuracy was examined to propose refinement of the evaluation criteria for the D_{max} index. Since D_{max} is an index of the strength of multidimensionality, its magnitude should be related to the classification accuracy. As a test exhibits more explicit multidimensionality, DETECT should be able to identify the dimensional structure of the test more easily. The evaluation criteria for evaluating D_{max} should be made in relation to the classification accuracy. Thus, a regression analysis between D_{max} and the classification accuracy was conducted to establish a new critical value for evaluating D_{max}. The classification

accuracy was treated as the independent variable while D_{max} was treated as the dependent variable.

The relationship between the r_{max} values and the classification accuracy was investigated using regression analysis. A clear positive relationship was found between the r_{max} values and the classification accuracy (Gierl et al., in press). According to Kim (1994), a r_{max} value less than 0.80 indicates complex data structure. As discussed previously in the description of the SAT composite data, the definition of simple or approximate simple structure might be too restrictive. A regression analysis was conducted to find a better break point to indicate simple or approximate simple structure, above which DETECT could identify the true dimensional structure of a test with ease. Since the degree of complexity in data structure and the correlation between dimensions might not be known to researchers in real data analysis, this break point in r_{max} index can serve as the critical value above which DETECT classification results are reliable. The classification accuracy was treated as independent variable while the r_{max} index was treated as the dependent variable.

MULTISIM. MULTISIM is a FORTRAN program that simulates dichotomous multidimensional test responses using the compensatory MIRT model. The maximum numbers of dimensions and items it can simulate are 4 and 120 respectively. It employs a user-specified multivariate normal distribution as the underlying latent ability distribution. The output from the program includes the simulated dataset, summary statistics of the dataset, and the ability estimates for the simulated sample.

Stage 2: Real Data Studies

One common critique of simulation studies is that the simulated conditions might not resemble real testing situations and, thus, have limited generalizability to real tests. This is because the item parameters used for simulation are usually systematically generated. However, since realistic item parameters were used for the simulation in the present study, there should be a close correspondence between the simulated and the real conditions. In order to establish the connection between the simulated conditions and real testing situations, real data analyses were conducted using the SAT 2005 March administration data.

Data

The SAT. Data from the 2005 March administration of the SAT were used in the current study. The SAT is a high-stake standardized test designed to measure college readiness. Both critical thinking and reasoning skills are tested. It includes three sections, Math, Critical Reading, and Writing. Only the Math and Critical Reading sections were used in the current study. This is because these two sections have been studied and hypotheses about their dimensional structure have been proposed and tested (Gierl et al., 2005). These hypotheses were used as the hypothesized true dimensional structure to be confirmed in the present study.

The Math section contains 54 items referenced to four primary skills and four content areas. The four skill categories are Applying Basic Mathematical Knowledge (AMK1), Applying Advanced Mathematical Knowledge (AMK2), Managing Complexity (MC), and Modeling and Insight (creating representation and insight, CRI) (O'Callaghan, Morley, & Schwartz, 2004). The four content areas are Algebra, Arithmetic, Geometry,

and Miscellaneous. Although both multiple-choice and constructed response items are included, both are scored dichotomously.

The Critical Reading section contains 67 items of two item types referenced to four primary skills. The four skill categories are Determining the Meaning of Words (WM), Understanding the Content, Form, and Function of Sentences (LC), Understanding the Content, Form, and Function of Larger Sections of Text (GC), and Analyzing Authors' Goals and Strategies (P) (VanderVeen, 2004). The two item types are sentence completion items and critical reading items associated with short and long passages. All items are multiple-choice items and are scored dichotomously. There are 121 items in total for the two subtests. All of the items were used in the data analysis.

Participants. A total of 294,960 students took the 2005 March administration of the SAT. These students, typically, are high school juniors in the U.S. as well as in Canada who intend to go to college in the U.S.

Samples. Two random samples of 2500 examinees were extracted from the data. The first sample served as the testing sample. The second sample served as the cross-validation sample.

Data Analyses

Three analyses were involved, one at the composite test level (Math and Critical Reading), and two at the subtest level (Math). The three datasets allowed for the testing of different proposed dimensional structures.

At the composite test level, the dataset included both the Math and the Critical Reading subtests that were identified as two separate dimensions by Gierl et al. (2005).

The composite test was expected to assume an approximate simple structure and to affirm the results of the simulated simple structure conditions.

At the subtest level, two analyses were conducted for the Math section. Two datasets were extracted and analyzed. Gierl et al. (2005) found that the AMK1 (skill 1) and the CRI (skill 4) items were distinct from each other while the AMK2 (skill 2) and MC (skill 3) items were not distinct. Thus, the skills 1 and 4 items were extracted to form the first dataset, and the skills 2 and 3 items were extracted to form the second dataset. Analysis of the first dataset was expected to yield fairly accurate and consistent results similar to the simulated low complexity condition (i.e., the complex 40% conditions), while analysis of the second dataset was expected to yield inaccurate and inconsistent results similar to the results of the simulated high complexity condition (i.e., the complex 80% conditions).

For all three analyses and the initial and cross-validation samples, NOHARM (Fraser, 1988) was used to obtain the item parameters and the correlation between the dimensions using an exploratory two dimensional compensatory MIRT model. Then, DETECT was used to obtain the D_{max} and r_{max} indices and the classification results. In order to get the classification accuracy and consistency, the two samples were analyzed with DETECT separately. The average of the classification accuracy was obtained as the final statistic; the match between the two samples was calculated as the classification consistency. The NOHARM results and the hypothesized truth of the dimensional structure not only provided information on the properties of the dataset but also helped to make the connection between the real data and the corresponding simulated condition.

NOHARM. NOHARM (Fraser, 1988), the acronym for the Normal Ogive Harmonic Analysis Robust Method, uses a nonlinear factor analytic approach (McDonald, 1967) to fit the unidimensional and multidimensional normal ogive models of latent trait to test data. The program can be used to model multidimensional data either using a nonlinear factor analytic model or an equivalent latent trait compensatory MIRT model.

NOHARM was used to obtain the correlation between dimensions and to estimate item parameters. Using the common factor parameterization of the multidimensional model, one can estimate the correlation between factors that represent dimensions underlying a test. Using the latent trait parameterization of the multidimensional model, one can estimate the multidimensional item parameters of the test items.

NOHARM can be used in either exploratory or confirmatory mode. When the underlying dimensional structure is not known, the exploratory mode should be used. However, if substantive analysis precedes statistical analysis or hypothesis about the dimensional structure of a test exists, the confirmatory mode should be used. However, when the confirmatory mode is adopted, the pattern matrix entered for specifying the dimensions that test items load on forces a simple structure for the data. Since complex dimensional structures are present in the datasets involved in the present study, using the confirmatory mode will not obtain accurate estimates of the item parameters. Thus, the exploratory mode of NOHARM was used.

Chapter 5: Results

The results of the analyses described in the previous chapter are presented in Chapter 5. The results from the simulation studies are presented first, followed by the results from the real data studies. The results from the real data studies are described with reference to the simulation results to establish connections between the simulated and real testing situations.

Simulation Results

The results from the simulation studies are presented in two sections, one for the results associated with the LSAT parameters, the other for the results associated with the SAT parameters. Due to the similar nature of the D_{max} and r_{max} results obtained for the initial and the cross validation samples, only the D_{max} and r_{max} results for the initial samples are presented. The classification accuracy, classification consistency, and misclassification error are presented for three subsets of the simulated datasets. First, results for the overall datasets are presented and discussed. Then, results are presented and discussed for items measuring dimensions one and two separately. Finally, results for the complex structure items are presented and discussed.

To evaluate whether the differences in D_{max}, r_{max}, classification accuracy, classification consistency, and misclassification error were significant across different levels within an independent variable, a critical value was developed. There were two reasons why a critical value was used to evaluate mean differences. First, the existence of three independent variables with three to four levels prohibited the use of ANOVA to analyze mean differences. It would be very difficult to disentangle the three-way interaction effect, let alone interpreting it. Moreover, a large number of post-hoc

comparisons would further complicate the analyses. Second, large sample size (100 replications) and small standard deviations across replications within conditions would make slight differences significant if *t*-tests were to be used. As shown in the previous study conducted by Gierl et al. (in press), the results from 100 replications within the studied conditions were relatively stable with standard deviations less than 0.10 for more than half of the cases. This would make mean differences less than 0.02 significant at the 0.05 level. Moreover, with a $3\times4\times3$ design, two sets of item parameters, and five statistics to be evaluated, there would be 750 comparisons to be made in the present study if consecutive levels within a factor were compared.

Based on these considerations, a critical value was derived from the effect size measure for mean differences. The effect size measure for mean differences is calculated with the following formula (Cohen, 1988):

$$d = \frac{|M_1 - M_2|}{\sqrt{\frac{n_1\sigma_1^2 + n_2\sigma_2^2}{n_1 + n_2}}},$$

where d is Cohen's effect size measure, M_1 and M_2 are the means for samples one and two, n_1 and n_2 are the sample sizes for samples one and two, and σ_1^2 and σ_2^2 are the variances for samples one and two. According to Cohen (1988), a d of 0.2 indicates a small effect, a d of 0.5 indicates a medium effect, and a d of 0.8 indicates a large effect. The 0.2 value was selected as indicative of significant differences in the present study. A review of the Gierl et al. (in press) study found that the standard deviations within conditions were all less than or equal to 0.25 with one exception. For that case the standard deviation was 0.26. Thus, a standard deviation of 0.25 for both samples was used to obtain the critical value for mean differences. When these values were substituted

into the formula, a mean difference of 0.05 was obtained. Thus, a critical value of 0.05 was used to evaluate mean differences in D_{max}, r_{max}, classification accuracy, classification consistency, and misclassification error.

The differences in values between levels within an independent variable, if greater than or equal to 0.05, were considered significant and indicative of meaningful impact. For example, suppose classification accuracy values of 0.90, 0.95, and 1.00 were obtained for the sample size conditions of 1500, 2000, and 2500, respectively. The differences across levels are both positive and at the critical value of 0.05. Thus, they indicate significant differences and a consistent positive effect of sample size on classification accuracy.

LSAT Results

D_{max} and r_{max} indices. The means and standard deviations of the D_{max} and r_{max} indices obtained for the different conditions simulated with the LSAT parameters are presented in Table 6. The standard deviations are presented in parenthesis. For the D_{max} index, as the degree of complexity increased, D_{max} decreased consistently with mean differences greater than or equal to 0.05 (0.05 to 0.32) for all cases except for the three cases where mean differences were 0.02 ($r = 0.90$, complex 40% vs. 80%). Degree of complexity generally showed a consistent negative impact on D_{max}. As the correlation between dimensions increased, D_{max} decreased consistently with mean differences greater than or equal to 0.05 (from 0.06 to 0.16) for the approximate simple and complex 40% conditions, whereas the mean differences across different levels of complexity were less than 0.05 for the complex 80% conditions. There was an interaction between correlation and degree of complexity. Correlation between dimensions showed a negative

Table 6

D_{max} and r_{max} Indices for Simulated Conditions with the LSAT Parameters

		Structure					
		Simple		Complex 40%		Complex 80%	
Correlation	Sample	D_{max}	r_{max}	D_{max}	r_{max}	D_{max}	r_{max}
0.60	1500	0.62 (0.03)	0.94 (0.02)	0.31 (0.02)	0.69 (0.03)	0.11 (0.01)	0.36 (0.04)
	2000	0.63 (0.03)	0.96 (0.02)	0.31 (0.02)	0.74 (0.03)	0.11 (0.01)	0.39 (0.04)
	2500	0.61 (0.03)	0.97 (0.03)	0.31 (0.02)	0.78 (0.02)	0.11 (0.01)	0.42 (0.04)
0.70	1500	0.46 (0.03)	0.87 (0.02)	0.23 (0.02)	0.60 (0.03)	0.09 (0.01)	0.32 (0.04)
	2000	0.47 (0.02)	0.92 (0.01)	0.24 (0.02)	0.65 (0.03)	0.09 (0.01)	0.33 (0.04)
	2500	0.46 (0.02)	0.94 (0.01)	0.24 (0.01)	0.69 (0.03)	0.09 (0.01)	0.37 (0.04)
0.80	1500	0.31 (0.02)	0.72 (0.03)	0.16 (0.02)	0.47 (0.04)	0.08 (0.01)	0.30 (0.03)
	2000	0.31 (0.02)	0.79 (0.03)	0.16 (0.01)	0.52 (0.03)	0.08 (0.01)	0.30 (0.03)
	2500	0.31 (0.02)	0.83 (0.02)	0.16 (0.02)	0.55 (0.04)	0.07 (0.01)	0.32 (0.03)
0.90	1500	0.15 (0.03)	0.42 (0.07)	0.10 (0.01)	0.32 (0.04)	0.08 (0.01)	0.29 (0.03)
	2000	0.16 (0.02)	0.51 (0.06)	0.09 (0.01)	0.34 (0.05)	0.07 (0.01)	0.29 (0.03)
	2500	0.15 (0.01)	0.53 (0.05)	0.09 (0.01)	0.36 (0.05)	0.07 (0.01)	0.30 (0.03)

impact on D_{max} only when the approximate simple or complex 40% structures were involved. There was no impact when the complex 80% structure was involved. However, when the results for the complex 80% conditions were examined, the D_{max} values clustered at the low end of the scale. In contrast to the approximate simple and complex 40% conditions where D_{max} ranged from 0.15 to 0.63 and from 0.09 to 0.31 respectively, for the complex 80% conditions D_{max} remained within the range of 0.07 to 0.11. The narrow range of values for the complex 80% conditions led to the interaction effect between correlation and degree of complexity. Sample size did not show an impact on

D_{max} : as sample size increased, the mean differences across levels of sample size were less than 0.05 for all cases.

For the r_{max} index, as the degree of complexity increased, r_{max} decreased consistently with mean differences greater than or equal to 0.05 (from 0.05 to 0.35) for all except one case with a mean difference of 0.03 ($r = 0.90$, $n = 1500$, Complex 40% vs. 80%). Thus, degree of complexity showed a negative impact on r_{max}.

As the correlation between dimensions increased, r_{max} decreased consistently with mean differences greater than or equal to 0.05 (from 0.07 to 0.30) for all except two cases (simple, $n = 2000, 2500$, 0.60 vs. 0.70) for the approximate simple and complex 40% conditions; r_{max} decreased consistently with mean differences greater than or equal to 0.05 (from 0.05 to 0.06) for only three of the nine cases for the complex 80% conditions ($n = 2000, 2500$, 0.60 vs. 0.70; $n = 2500$, 0.70 vs. 0.80). An interaction effect was again found between correlation and degree of complexity. Correlation between dimensions showed a negative impact on r_{max} only when approximate simple or complex 40% structures were involved. When the results were examined, for the two cases in the approximate simple structure conditions with mean differences less than 0.05, the r_{max} values were very close to the highest possible value of 1, which led to the insignificant mean differences. However, for the complex 80% conditions, the r_{max} values clustered at the low end. Compared with the ranges of 0.42 to 0.97 and 0.32 to 0.78 for the approximate simple and complex 40% conditions, r_{max} had a narrow range from 0.29 to 0.42 for the complex 80% conditions. This explained the interaction effect between correlation and degree of complexity.

As sample size increased, r_{max} increased consistently with mean differences greater than or equal 0.05 (from 0.05 to 0.09) only for six of 24 cases (simple, $r = 0.70$, 0.80, 0.90, 1500 vs. 2000; complex 40%, $r = 0.60$, 0.70, 0.80, 1500 vs. 2000). All six cases occurred when sample size increased from 1500 to 2000. However, the significant results were not across all four correlation conditions for the approximate simple and complex 40% conditions. This could be attributable to the clustering of values at both ends of the scale: for the approximate simple and 0.60 correlation conditions, the r_{max} values clustered within the range of 0.94 to 0.97; and for the complex 40% and 0.90 conditions, the r_{max} values clustered within the range of 0.32 to 0.36. Despite these narrow ranges of r_{max} values, sample size showed a positive impact on r_{max} when it increased from 1500 to 2000 for the approximate simple and complex 40% conditions.

Based on the evaluation criteria for the two indices (Kim, 1994), the three cases in the complex 40% conditions for the correlation of 0.90 and the nine cases in the complex 80% conditions for correlations of 0.70, 0.80, and 0.90 were judged as essentially unidimensional. For all these cases, the obtained D_{max} values were less than or equal to 0.10. All other conditions were judged as displaying weak ($0.1 < D_{max} \leq 0.5$) to moderate multidimensionality ($0.5 < D_{max} \leq 1$). Further, the r_{max} indices obtained for all the complex structure conditions were less than 0.80—the evaluation criterion for r_{max}. However, five cases in the approximate simple structure conditions (correlation 0.80 conditions with sample sizes of 1500 and 2000 and all three correlation 0.90 conditions) were judged as having a complex structure with r_{max} values less than 0.80. The standard

deviations of D_{max} and r_{max} across 100 replications were relatively small (less than or equal to 0.05) indicating stability of results across replications.

Classification accuracy (overall). The classification accuracy results are presented in three tables. Table 7 contains the means and standard deviations of overall classification accuracy results when all 40 items were considered in the simulated datasets. The standard deviations are in parenthesis.

Table 7

Overall Classification Accuracy for Simulated Conditions with the LSAT Parameters

		Structure		
Correlation	Sample	Simple	Complex 40%	Complex 80%
0.60	1500	1.00 (0.01)	0.96 (0.03)	0.77 (0.09)
	2000	1.00 (0.01)	0.98 (0.02)	0.84(0.08)
	2500	1.00 (0.01)	0.99 (0.02)	0.88 (0.06)
0.70	1500	1.00 (0.01)	0.94 (0.03)	0.70 (0.10)
	2000	1.00 (0.00)	0.96 (0.03)	0.75 (0.10)
	2500	1.00 (0.00)	0.98 (0.02)	0.81 (0.08)
0.80	1500	1.00 (0.01)	0.89 (0.05)	0.61 (0.10)
	2000	1.00 (0.01)	0.92 (0.04)	0.65 (0.09)
	2500	1.00 (0.00)	0.95 (0.05)	0.71 (0.10)
0.90	1500	0.92 (0.14)	0.70 (0.12)	0.56 (0.08)
	2000	0.99 (0.05)	0.75 (0.12)	0.58 (0.08)
	2500	0.99 (0.05)	0.81 (0.12)	0.60 (0.07)

The criterion for acceptable accuracy was set at 0.85 (i.e., 85% of items accurately placed on the dimensions they were simulated to measure) in this study. Classification accuracy should be fairly high for meaningful interpretation of the dimensions identified. A classification accuracy of 85% indicates that no more than six items should be misclassified for a 40-item test. A criterion of 85% is strict but reasonable.

As indicated in Table 7, overall classification accuracy was above criterion for all approximate simple structure conditions (92% to 100%). For the complex 40% conditions, overall classification accuracy was above criterion for all but the three cases when the correlation was 0.90. For the complex 80% conditions, overall classification accuracy was above criterion only for the condition with a correlation of 0.60 and a sample size of 2500.

As the degree of complexity increased, classification accuracy decreased consistently with mean differences greater than or equal to 5% (from 5% to 28%) for all except five cases ($r = 0.60$, $n = 1500, 2000, 2500$, simple vs. complex 40%; $r = 0.7$, $n = 2000, 2500$, simple vs. complex 40%). All exceptions occurred for the two lower values of correlation. When the results were examined, for the approximate simple and the lower correlation (0.60 and 0.70) conditions for the complex 40% conditions, classification accuracy was close to 100%, the highest possible value. This led to the five exceptions with mean differences less than 5%. Thus, degree of complexity showed a negative impact on classification accuracy.

As the correlation between dimensions increased, for the approximate simple structure conditions, classification accuracy decreased consistently with mean differences greater than or equal to 5% (7%) only for one case ($n = 1500$, 0.80 vs. 0.90). For the complex 40% conditions, classification accuracy decreased consistently with mean differences from 5% to 19% for four of nine cases. Three of the four cases occurred for the comparison between the 0.80 and the 0.90 correlation conditions. For the complex 80% conditions, classification accuracy decreased consistently with mean differences greater than or equal to 5% (5% to 11%) for all cases. Thus, there was an interaction

effect between correlation and degree of complexity: correlation between dimensions did not show an impact on classification accuracy for the approximate simple structure conditions, a negative impact when the correlation went from 0.80 to 0.90 for the complex 40% conditions, and a negative impact across all levels of correlation for the complex 80% condition. Again, the clustering of values close to 100% for the approximate simple and the two lower correlation conditions of the complex 40% conditions led to the interaction effect between correlation and degree of complexity.

As sample size increased, classification accuracy increased consistently with mean differences greater than or equal to 5% (from 5% to 7%) for seven of 24 cases (simple, $r = 0.90$, 1500 vs. 2000; complex 40%, $r = 0.90$, 1500 vs. 2000, and 2000 vs. 2500; complex 40%, $r = 0.60$, 1500 vs. 2000, $r = 0.70$, 1500 vs. 2000, 2000 vs. 2500, and $r = 0.80$, 2000 vs. 2500). These cases spread across all conditions without an identifiable pattern. Thus, sample size did not show a significant impact on classification accuracy.

Classification accuracy (dimensions one and two items). Table 8 contains the classification accuracy results for dimensions one and two. These results were obtained to check if classification error occurred more often in one of the two dimensions. The mean differences between dimensions one and two were also evaluated with the 0.05 critical value. The results showed that for the approximate simple and the complex 40% conditions the classification accuracy results for the two dimensions were similar to one another with mean differences less than 5%. However, for the complex 80% conditions, the classification accuracy results for the dimension one items were consistently higher than those for the dimension two items for seven of 12 cases. The impact of the three independent variables—degree of complexity, correlation between dimensions, and

sample size—and the pattern of results (above or below criterion) were the same as for the overall classification accuracy results.

Table 8

Classification Accuracy for Dimensions One and Two for Simulated Conditions with the LSAT Parameters

		Structure					
		Simple		Complex 40%		Complex 80%	
Correlation	Sample	Dim. I	Dim. II	Dim. I	Dim. II	Dim. I	Dim. II
0.60	1500	1.00 (0.01)	1.00 (0.00)	0.96 (0.04)	0.96 (0.04)	0.79 (0.13)	0.75 (0.12)
	2000	1.00 (0.01)	1.00 (0.01)	0.98 (0.03)	0.98 (0.03)	0.87 (0.11)	0.80 (0.11)
	2500	1.00 (0.01)	1.00 (0.00)	0.98 (0.03)	1.00 (0.02)	0.92 (0.08)	0.84 (0.08)
0.70	1500	1.00 (0.01)	1.00 (0.01)	0.94 (0.06)	0.94 (0.05)	0.71 (0.16)	0.68 (0.12)
	2000	1.00 (0.00)	1.00 (0.00)	0.96 (0.05)	0.96 (0.05)	0.77 (0.17)	0.74 (0.13)
	2500	1.00 (0.00)	1.00 (0.00)	0.97 (0.04)	0.99 (0.03)	0.89 (0.08)	0.73 (0.14)
0.80	1500	1.00 (0.01)	1.00 (0.01)	0.90 (0.07)	0.88 (0.09)	0.63 (0.18)	0.59 (0.12)
	2000	1.00 (0.01)	1.00 (0.01)	0.92 (0.07)	0.92 (0.07)	0.70 (0.16)	0.60 (0.15)
	2500	1.00 (0.01)	1.00 (0.00)	0.94 (0.06)	0.96 (0.08)	0.80 (0.13)	0.62 (0.16)
0.90	1500	0.92 (0.12)	0.91 (0.17)	0.71 (0.16)	0.68 (0.16)	0.56 (0.16)	0.56 (0.11)
	2000	0.98 (0.06)	0.99 (0.05)	0.76 (0.17)	0.75 (0.16)	0.61 (0.17)	0.55 (0.12)
	2500	0.99 (0.03)	0.99 (0.07)	0.82 (0.16)	0.81 (0.17)	0.70 (0.15)	0.50 (0.11)

Classification accuracy (complex structure items). The classification accuracy results for the subsets of complex structure items are presented in Table 9. The classification accuracy results for these items were lower than the overall classification accuracy results (Table 7) with mean differences greater than or equal 5% for all except three cases for the complex 40% conditions ($r = 0.60$, $n = 2000, 2500$; $r = 0.70$, $n = 2500$). For the complex 80% conditions, the classification accuracy results for the

complex items were comparable to the overall classification accuracy except for two cases with mean differences of 5% ($r = 0.70$, $n = 1500$; $r = 0.80$, $n = 2000$). The complex structure items measured a composite of dimensions one and two, which makes it difficult for DETECT to classify them correctly. This likely led to the lower classification accuracy results for the complex structure items for the complex 40% conditions. However, for the complex 80% conditions, a large proportion of items (32 of 40) were complex structure items, making the classification accuracy results for the subsets of complex item comparable to those for the overall datasets.

Table 9

Classification Accuracy for Complex Structure Items for Simulated Conditions with the LSAT Parameters

		Complex Structure	
Correlation	Sample	40% (16 items)	80% (32 items)
0.60	1500	0.91 (0.06)	0.73 (0.10)
	2000	0.96 (0.05)	0.80 (0.09)
	2500	0.97 (0.04)	0.85 (0.07)
0.70	1500	0.84 (0.08)	0.65 (0.10)
	2000	0.90 (0.08)	0.71 (0.11)
	2500	0.95 (0.06)	0.77 (0.10)
0.80	1500	0.74 (0.10)	0.57 (0.10)
	2000	0.81 (0.11)	0.60 (0.10)
	2500	0.88 (0.09)	0.67 (0.11)
0.90	1500	0.55 (0.11)	0.53 (0.08)
	2000	0.59 (0.12)	0.55 (0.09)
	2500	0.67 (0.15)	0.57 (0.08)

Classification consistency and misclassification error (overall). High consistency among samples is needed to gain confidence in DETECT classification results when the true dimensional structure is not known. A cross-validation sample was generated for each simulated condition to calculate the classification consistency and misclassification

error rate in the present study. The classification consistency and misclassification error rate were calculated for the overall tests, the dimension one and dimension two items separately, and the complex structure items. The same evaluation criterion of 85% was used for evaluating classification consistency. An index of error rate, misclassification error should be fairly low for classification consistency results to be useful for inferring the dimensional structure of a test. Therefore, the criterion of 5% was used for evaluating misclassification error: misclassification error values 5% or greater were considered to be significantly different from zero.

Table 10 includes the overall classification consistency and misclassification error results. Classification consistency was above the criterion of 85% for all the approximate simple structure conditions. For the complex 40% conditions, classification consistency was above or on criterion when the correlation between dimensions was 0.80 or lower. For the complex 80% conditions, none of the classification consistency results were above criterion.

As the degree of complexity increased, classification consistency decreased consistently with mean differences greater than or equal to 5% (from 0.07 to 0.30) for all except two cases ($r = 0.60$, $n = 2000$, 2500, simple vs. complex 40%). Degree of complexity showed a negative impact on classification consistency. Clustering at the high end of the scale was also found for classification consistency. Classification consistency was at the highest possible value of 100% for the approximate simple structure conditions with correlations of 0.80 or lower and was close to 100% for the lower correlation conditions (0.60 and 0.70) for the complex 40% conditions. This likely led to the two

cases with mean differences smaller than 5% for the approximate simple structure conditions with a correlation of 0.60.

Table 10

Classification Consistency and Misclassification Error Rates between Primary and Cross-Validation Samples for Simulated Conditions with the LSAT Parameters

		Structure					
		Simple		Complex 40%		Complex 80%	
Correlation	Sample	CC	ME	CC	ME	CC	ME
0.60	1500	1.00 (0.01)	0.00 (0.00)	0.93 (0.04)	0.00 (0.01)	0.69 (0.11)	0.07 (0.04)
	2000	1.00 (0.01)	0.00 (0.00)	0.97 (0.03)	0.00 (0.01)	0.74 (0.11)	0.05 (0.04)
	2500	1.00 (0.01)	0.00 (0.00)	0.98 (0.02)	0.00 (0.00)	0.84 (0.09)	0.05 (0.03)
0.70	1500	1.00 (0.01)	0.00 (0.00)	0.90 (0.04)	0.01 (0.02)	0.63 (0.11)	0.12 (0.05)
	2000	1.00 (0.01)	0.00 (0.00)	0.93 (0.04)	0.01 (0.01)	0.66 (0.12)	0.09 (0.06)
	2500	1.00 (0.00)	0.00 (0.00)	0.96 (0.03)	0.00 (0.01)	0.76 (0.10)	0.09 (0.05)
0.80	1500	1.00 (0.01)	0.00 (0.00)	0.85 (0.06)	0.03 (0.03)	0.55 (0.14)	0.17 (0.08)
	2000	1.00 (0.01)	0.00 (0.00)	0.87 (0.05)	0.02 (0.02)	0.58 (0.13)	0.15 (0.07)
	2500	1.00 (0.01)	0.00 (0.00)	0.92 (0.06)	0.01 (0.01)	0.70 (0.13)	0.15 (0.08)
0.90	1500	0.88 (0.15)	0.00 (0.01)	0.64 (0.13)	0.11 (0.06)	0.53 (0.13)	0.21 (0.09)
	2000	0.95 (0.12)	0.00 (0.00)	0.70 (0.13)	0.09 (0.06)	0.52 (0.14)	0.19 (0.09)
	2500	0.99 (0.05)	0.00 (0.00)	0.73 (0.14)	0.06 (0.05)	0.61 (0.17)	0.22 (0.08)

Note. CC is classification consistency, and ME is misclassification error.

As the correlation between dimensions increased, classification consistency decreased with mean differences from 5% to 12% for only two of the nine approximate simple cases (n = 1500, 2000, 0.80 vs. 0.90). For the complex 40% conditions,

classification consistency decreased consistently as the correlation increased with mean differences greater than or equal to 5% (5% to 21%) for five of the nine cases. Mean differences between correlations of 0.70 and 0.80 were meaningful for two sample size conditions of 1500 and 2000; mean differences between correlations of 0.80 and 0.90 were meaningful for all three sample size conditions. For the complex 80% conditions, classification consistency decreased consistently as the correlation increased with mean differences greater than or equal to 5% (6% to 9%) for all but one of the nine cases (n = 1500, 0.80 vs. 0.90). Thus, an interaction effect was found between correlation and degree of complexity: correlation did not show an impact on classification consistency for the approximate simple structure conditions; a negative impact when it went from 0.70 to 0.90 for the complex 40% conditions; and a consistent negative impact across all levels of correlation for the complex 80% conditions. The clustering of values close to 100% accounted for the interaction between correlation and degree of complexity.

As sample size increased, classification consistency increased consistently with mean differences greater than or equal to 5% for eight of 24 cases. For the approximate simple and complex 40% conditions, three cases of mean differences above 5% (from 5% to 12%) spread across all conditions (simple, r = 0.90, 1500 vs. 2000; complex 40%, r = 0.80, 2000 vs. 2500, r = 0.90, 1500 vs. 2000). For the complex 80% conditions, four cases of mean differences above 5% occurred when sample size went from 2000 to 2500 across all correlation conditions. Thus, sample size showed a positive effect on classification consistency when it went from 2000 to 2500 for the complex 80% conditions.

The misclassification error rate was zero for all approximate simple structure conditions; DETECT not only consistently classified items into the same dimensions across samples but also identified the correct dimensions for each item. For the complex 40% conditions, the misclassification error rate was satisfactory (less than or equal to 5%) when the correlation between dimensions was less than or equal to 0.80. For the complex 80% conditions, the misclassification error rate was satisfactory only when the correlation between dimensions was 0.60 and the sample size was 2000 or higher.

As the degree of complexity increased, the misclassification error rate increased consistently with mean differences greater than or equal to 5% (from 5% to 16%) for 15 of 24 cases. For the correlations of 0.60, 0.70, and 0.80, misclassification error increased with mean differences greater or equal to 5% when degree of complexity went from 40% to 80%. For the correlation of 0.90, misclassification error increased with meaningful mean differences for all cases. Therefore, an interaction effect was found between degree of complexity and correlation. Degree of complexity showed a positive impact on misclassification error when it went from 40% to 80% for the correlations of 0.60 to 0.80. For the correlation of 0.90, it showed a positive impact on misclassification error across all levels of complexity. When the results were examined, misclassification error rates clustered near the lowest possible value of 0% for the low complexity (simple and complex 40%) conditions. This accounted for the interaction effect between degree of complexity and correlation between dimensions.

As the correlation between dimensions increased, misclassification error increased consistently with mean differences greater than or equal to 5% (from 5% to 8%) for eight of 27 cases. For the approximate simple structure conditions, misclassification

error remained zero. For the complex 40% conditions, mean differences above 5% occurred for all three sample size conditions when the correlation went from 0.80 to 0.90. For the complex 80% conditions, mean differences above 5% occurred for all three sample size conditions when the correlation went from 0.70 to 0.80. Thus, there was an interaction effect between correlation and degree of complexity. For the approximate simple conditions, correlation did not show an impact on misclassification error. For the complex 40% conditions, correlation showed a positive impact when the correlation went from 0.80 to 0.90. For the complex 80% conditions, correlation showed a positive impact when correlation went from 0.70 to 0.80.

As sample size increased, mean differences in misclassification error were all below 5%. Thus, sample size did not show a significant impact on misclassification error.

Classification consistency and misclassification error (dimensions one and two items and complex items). Table 11 contains the classification consistency and misclassification error results for the dimension one and dimension two items separately, and Table 12 contains the classification consistency and misclassification error results for the complex structure items. The classification consistency results for the dimension one and dimension two items were similar for the approximate simple and complex 40% conditions with mean differences less than 5% for all cases. The results for the dimension one items were higher than those for the dimension two items for six of 12 cases for the complex 80% conditions ($r = 0.60$, $n = 2000, 2500$; $r = 0.70$, $n = 2000, 2500$; $r = 0.80$, $n = 2000, 2500$).

The misclassification error rates for the dimensions one and two items were similar for the approximate simple and complex 40% conditions with mean differences

Table 11

Classification Consistency and Misclassification Error Rates for Dimensions One and Two for the LSAT Parameters

		Structure											
		Simple				Complex 40%				Complex 80%			
		Dim. I		Dim. II		Dim. I		Dim. II		Dim. I		Dim. II	
Correlation	Sample	CC	ME	CC	ME	CC	ME	CC	ME	CC	ME	CC	ME
0.60	1500	1.00 (0.01)	0.00 (0.00)	1.00 (0.00)	0.00 (0.00)	0.93 (0.06)	0.00 (0.01)	0.94 (0.05)	0.00 (0.01)	0.69 (0.14)	0.05 (0.05)	0.68 (0.11)	0.09 (0.07)
	2000	0.99 (0.02)	0.00 (0.00)	1.00 (0.01)	0.00 (0.00)	0.97 (0.04)	0.00 (0.01)	0.97 (0.04)	0.00 (0.01)	0.79 (0.12)	0.03 (0.05)	0.69 (0.13)	0.07 (0.06)
	2500	0.99 (0.02)	0.00 (0.00)	1.00 (0.00)	0.00 (0.00)	0.97 (0.04)	0.00 (0.01)	0.99 (0.02)	0.00 (0.01)	0.88 (0.11)	0.02 (0.02)	0.81 (0.10)	0.07 (0.06)
0.70	1500	1.00 (0.02)	0.00 (0.00)	1.00 (0.01)	0.00 (0.00)	0.90 (0.07)	0.01 (0.03)	0.90 (0.07)	0.01 (0.03)	0.64 (0.13)	0.10 (0.10)	0.61 (0.14)	0.13 (0.08)
	2000	1.00 (0.01)	0.00 (0.00)	1.00 (0.00)	0.00 (0.00)	0.93 (0.06)	0.01 (0.02)	0.94 (0.06)	0.01 (0.02)	0.68 (0.14)	0.07 (0.09)	0.63 (0.13)	0.11 (0.09)
	2500	1.00 (0.01)	0.00 (0.00)	1.00 (0.00)	0.00 (0.00)	0.95 (0.05)	0.00 (0.01)	0.98 (0.04)	0.00 (0.01)	0.82 (0.13)	0.04 (0.04)	0.71 (0.12)	0.14 (0.09)
0.80	1500	1.00 (0.02)	0.00 (0.00)	1.00 (0.01)	0.00 (0.00)	0.86 (0.07)	0.03 (0.03)	0.85 (0.09)	0.04 (0.04)	0.55 (0.16)	0.15 (0.12)	0.56 (0.15)	0.18 (0.10)
	2000	1.00 (0.01)	0.00 (0.00)	1.00 (0.02)	0.00 (0.00)	0.88 (0.08)	0.01 (0.03)	0.87 (0.07)	0.02 (0.04)	0.61 (0.16)	0.11 (0.10)	0.56 (0.14)	0.18 (0.12)
	2500	1.00 (0.01)	0.00 (0.00)	1.00 (0.01)	0.00 (0.00)	0.90 (0.07)	0.01 (0.02)	0.94 (0.08)	0.01 (0.02)	0.75 (0.16)	0.08 (0.06)	0.64 (0.12)	0.22 (0.13)
0.90	1500	0.88 (0.13)	0.00 (0.01)	0.88 (0.18)	0.00 (0.02)	0.65 (0.15)	0.10 (0.08)	0.63 (0.14)	0.12 (0.10)	0.53 (0.16)	0.22 (0.15)	0.52 (0.15)	0.21 (0.10)
	2000	0.95 (0.10)	0.00 (0.00)	0.95 (0.16)	0.00 (0.00)	0.70 (0.15)	0.08 (0.10)	0.69 (0.14)	0.10 (0.09)	0.52 (0.16)	0.18 (0.14)	0.52 (0.15)	0.21 (0.13)
	2500	0.99 (0.03)	0.00 (0.01)	0.99 (0.07)	0.00 (0.00)	0.72 (0.15)	0.06 (0.08)	0.74 (0.16)	0.05 (0.07)	0.62 (0.20)	0.12 (0.09)	0.60 (0.17)	0.31 (0.15)

Note. CC is classification consistency, and ME is misclassification error.

Table 12

Classification Consistency and Misclassification Error Rates for Complex Structure Items for Simulated Conditions with the LSAT Parameters

		Complex Structure			
		40% (16 items)		80% (32 items)	
Correlation	Sample	CC	ME	CC	ME
0.60	1500	0.84 (0.09)	0.01 (0.02)	0.63 (0.12)	0.08 (0.05)
	2000	0.92 (0.07)	0.00 (0.01)	0.69 (0.12)	0.06 (0.05)
	2500	0.95 (0.05)	0.00 (0.01)	0.81 (0.10)	0.06 (0.04)
0.70	1500	0.75 (0.10)	0.03 (0.05)	0.58 (0.12)	0.14 (0.07)
	2000	0.83 (0.09)	0.02 (0.04)	0.61 (0.13)	0.10 (0.07)
	2500	0.91 (0.07)	0.01 (0.02)	0.72 (0.12)	0.10 (0.06)
0.80	1500	0.65 (0.11)	0.08 (0.06)	0.52 (0.14)	0.19 (0.09)
	2000	0.69 (0.12)	0.05 (0.05)	0.55 (0.14)	0.18 (0.08)
	2500	0.81 (0.11)	0.02 (0.04)	0.66 (0.13)	0.17 (0.09)
0.90	1500	0.54 (0.15)	0.21 (0.11)	0.51 (0.14)	0.24 (0.10)
	2000	0.55 (0.19)	0.19 (0.12)	0.50 (0.14)	0.22 (0.09)
	2500	0.57 (0.18)	0.13 (0.11)	0.59 (0.17)	0.24 (0.09)

Note. CC is classification consistency, and ME is misclassification error.

less than 5% for all cases. For the complex 80% conditions, misclassification error rates were lower for the dimension one items than those for the dimension two items with mean differences greater than or equal to 0.05 for five of 12 cases ($r = 0.60$, $n = 2500$; $r = 0.70$, $n = 2500$; $r = 0.80$, $n = 2000, 2500$; $r = 0.90$, $n = 2500$).

As seen from Table 12, the classification consistency for the complex structure items was lower than the overall classification consistency for all items (Table 10) with mean differences greater than or equal to 5% for all but one cases for the complex 40% conditions ($r = 0.60$, $n = 2500$). The classification consistency results for the complex

structure items were lower than the overall classification consistency results with mean differences greater than or equal to 5% for only four of 12 cases (r = 0.60, n = 1500, 2000; r = 0.70, n = 1500, 2000). On the other hand, the misclassification error rates for the complex structure items were higher than the overall misclassification error rates with mean differences greater than or equal to 5% for four of 12 cases for the complex 40% conditions (r = 0.80, n = 1500; r = 0.90, n = 1500, 2000, and 2500). For the complex 80% conditions, the misclassification error rates were comparable between those for the complex item and those for all the items with mean differences less than 5% for all cases.

SAT Results

D_{max} and r_{max} indices. The D_{max} and r_{max} indices obtained for different conditions simulated with the SAT parameters are presented in Table 13. As the degree of complexity increased, D_{max} decreased consistently with mean differences greater than or equal to 0.05 (from 0.06 to 0.07) only for three of 24 cases (r = 0.60, n = 1500, 2000, and 2500, simple vs. complex 40%). Thus, degree of complexity did not show a significant impact on D_{max}. In the case of the SAT, the D_{max} values clustered within the range of 0.07 to 0.19 for all cases. Given the relatively narrow range of D_{max} values, changes in the correlation (with one exception: r = 0.60, n = 1500, 0.60 vs. 0.70) and sample size did not significantly impacted D_{max}.

For the r_{max} index, as the degree of complexity increased, r_{max} decreased consistently with mean differences greater than or equal to 0.05 (from 0.05 to 0.18) for 10 of 24 cases. For the correlation of 0.60, r_{max} decreased with mean differences from 0.07

Table 13

D_{max} and r_{max} Indices for Simulated Conditions with the SAT Parameters

		Structure					
		Simple		Complex 40%		Complex 80%	
Correlation	Sample	D_{max}	r_{max}	D_{max}	r_{max}	D_{max}	r_{max}
0.60	1500	0.19 (0.02)	0.50 (0.05)	0.12 (0.02)	0.35 (0.04)	0.09 (0.01)	0.28 (0.03)
	2000	0.18 (0.03)	0.52 (0.09)	0.12 (0.01)	0.37 (0.04)	0.08 (0.01)	0.28 (0.03)
	2500	0.17 (0.02)	0.55 (0.07)	0.10 (0.01)	0.37 (0.05)	0.08 (0.01)	0.30 (0.03)
0.70	1500	0.14 (0.02)	0.39 (0.06)	0.11 (0.01)	0.32 (0.04)	0.09 (0.01)	0.28 (0.03)
	2000	0.14 (0.02)	0.42 (0.07)	0.10 (0.01)	0.33 (0.04)	0.08 (0.01)	0.28 (0.02)
	2500	0.13 (0.02)	0.43 (0.08)	0.09 (0.01)	0.33 (0.04)	0.08 (0.01)	0.30 (0.03)
0.80	1500	0.11 (0.01)	0.32 (0.04)	0.10 (0.01)	0.30 (0.03)	0.09 (0.01)	0.28 (0.03)
	2000	0.10 (0.01)	0.33 (0.04)	0.09 (0.01)	0.31 (0.03)	0.08 (0.01)	0.29 (0.02)
	2500	0.09 (0.01)	0.33 (0.04)	0.08 (0.01)	0.30 (0.03)	0.08 (0.01)	0.30 (0.03)
0.90	1500	0.10 (0.01)	0.29 (0.02)	0.10 (0.01)	0.29 (0.03)	0.09 (0.01)	0.28 (0.03)
	2000	0.09 (0.01)	0.31 (0.02)	0.08 (0.01)	0.29 (0.03)	0.08 (0.01)	0.29 (0.03)
	2500	0.08 (0.01)	0.31 (0.03)	0.08 (0.01)	0.30 (0.02)	0.07 (0.01)	0.29 (0.03)

to 0.18 for all cases. For the correlation of 0.70, four of six cases had mean differences from 0.05 to 0.10. Mean differences were above 0.05 for all three sample size conditions when degree of complexity went from 0% to 40%, whereas mean difference was 0.05 only for the sample size of 2000 when degree of complexity went from 40% to 80%. For the correlations of 0.80 and 0.90, mean differences were all less than 0.05. Thus, there was an interaction effect between degree of complexity and correlation. Degree of complexity showed a negative impact on r_{max} for the correlation of 0.60, while it showed

a negative impact on r_{max} for the correlation 0.70 when the degree of complexity went from 0% to 40%. Degree of complexity did not show an impact on r_{max} for the correlations of 0.80 and 0.90. The r_{max} values clustered within the range of 0.28 to 0.32 for the high correlation conditions (0.80 and 0.90) of the complex 40% conditions and for all complex 80% conditions. This accounted for the interaction effect between degree of complexity and correlation.

As the correlation between dimensions increased, r_{max} decreased for six of 27 cases. These cases all occurred for the approximate simple structure conditions when the correlation went from 0.6 to 0.7 and from 0.7 to 0.8. Therefore, an interaction effect was found between correlation and degree of complexity. Correlation between dimensions showed a negative impact on r_{max} for the approximate simple structure conditions when the correlation went from 0.6 to 0.8, but it did not show an impact on r_{max} for the complex 40% and complex 80% conditions. As sample size increased, mean differences in r_{max} were all less than 0.05. Thus, sample size did not show a significant impact on r_{max}.

Based on the evaluation criteria for the two indices (Kim, 1994), all cases in the complex 80% conditions were judged as unidimensional with D_{max} values less than or equal to 0.10. For the complex 40% conditions, nine of 12 cases were judged as unidimensional (when correlation was 0.60 and sample size was 2500, when correlation was 0.70 and sample size was 2000 or 2500, and when correlation was 0.80 or 0.90) and five cases in the approximate simple structure conditions (when correlation was 0.80 and sample size was 2000 or 2500, and when correlation was 0.90). All other cases displayed

weak multidimensionality. However, despite these differences, all conditions across the three independent variables were judged as having complex structures with r_{max} values less than 0.80. The standard deviations across 100 replications were small (less than or equal to 0.08) indicating stability across replications.

Classification accuracy (overall). The classification accuracy results are presented in three tables. The same evaluation criterion of 85% was used. As indicated in Table 14, overall classification accuracy was above criterion (86% to 95%) for the approximate simple structure conditions only for the correlation of 0.60 and for the correlation of 0.70 with sample sizes of 1500 and 2000. For the complex structure conditions, the overall classification accuracy was below criterion (50% to 80%) across all conditions.

Table 14

Overall Classification Accuracy for Simulated Conditions with the SAT Parameters

		Structure		
Correlation	Sample	Simple	Complex 40%	Complex 80%
0.60	1500	0.95 (0.05)	0.74 (0.10)	0.54 (0.10)
	2000	0.93 (0.15)	0.80 (0.09)	0.54 (0.12)
	2500	0.95 (0.12)	0.78 (0.13)	0.59 (0.12)
0.70	1500	0.86 (0.15)	0.68 (0.11)	0.55 (0.10)
	2000	0.86 (0.18)	0.73 (0.12)	0.55 (0.11)
	2500	0.82 (0.23)	0.70 (0.13)	0.57 (0.10)
0.80	1500	0.71 (0.17)	0.62 (0.09)	0.52 (0.09)
	2000	0.67 (0.19)	0.65 (0.12)	0.50 (0.11)
	2500	0.61 (0.19)	0.60 (0.12)	0.58 (0.11)
0.90	1500	0.58 (0.10)	0.59 (0.07)	0.51 (0.10)
	2000	0.53 (0.10)	0.59 (0.09)	0.52 (0.11)
	2500	0.55 (0.10)	0.54 (0.08)	0.56 (0.11)

As the degree of complexity increased, classification accuracy decreased with mean differences greater than or equal to 5% (from 12% to 26%) for the correlations of

0.60 and 0.70. Mean differences were from 9% to 15% for three of six cases for the correlation of 0.80 (n = 1500, simple vs. complex 40%; n = 1500, 2500, complex 40% vs. complex 80%). Two of these three cases occurred when the degree of complexity went from 40% to 80%. For the 0.90 correlation conditions, classification accuracy showed inconsistent change with one significant negative mean difference of 6% (n = 2000, simple vs. complex 40%) and two significant positive mean differences of 7% to 8% (n = 1500, 2000, complex 40% vs. complex 80%). Therefore, an interaction effect was found between degree of complexity and correlation: degree of complexity had a negative impact on classification accuracy for correlations of 0.60 and 0.70, a negative impact for the correlation of 0.80 when degree of complexity went from 40% to 80%, and an inconsistent impact for the correlation of 0.90. Classification accuracy clustered at the low end of the scale with a narrow range of values for the two high correlations (0.80 and 0.90) and the complex 80% conditions. This led to the interaction effect.

As the correlation between dimensions increased, classification accuracy decreased consistently with mean differences greater than or equal to 5% (from 6% to 21%) for all cases for the approximate simple structure conditions and all but one for the complex 40% conditions (complex 40%, n = 1500, 0.80 vs. 0.90). For the complex 80% conditions, only one case with mean difference greater than 5% was found (n = 2000, 0.70 vs. 0.80). Thus, there was an interaction effect between correlation and degree of complexity. Correlation between dimensions had a negative impact on classification accuracy for the approximate simple and complex 40% conditions and no impact for the complex 80% conditions. Again the clustering of classification accuracy values at the low end of the scale for the complex 80% conditions led to the interaction effect.

As sample size increased, inconsistent mean differences were found with four significant negative mean differences of 5% to 6% (simple, $r = 0.80$, 2000 vs. 2500, $r = 0.90$, 1500 vs. 2000; complex 40%, $r = 0.80$, 2000 vs. 2500, $r = 0.90$, 2000 vs. 2500) and four significant positive mean differences of 5% to 8% (complex 40%, $r = 0.60$, 1500 vs. 2000, $r = 0.70$, 1500 vs. 2000; complex 80%, $r = 0.60$, 2000 vs. 2500, $r = 0.80$, 2000 vs. 2500). Thus, sample size did not show a consistent impact on classification accuracy.

Classification accuracy (dimensions one and two items). Table 15 contains the classification accuracy results for the dimensions one and two items. These results were

Table 15

Classification Accuracy for Dimensions One and Two for Simulated Conditions with the SAT Parameters

		Structure					
		Simple		Complex 40%		Complex 80%	
Correlation	Sample	Dim. I	Dim. II	Dim. I	Dim. II	Dim. I	Dim. II
0.60	1500	0.98 (0.05)	0.92 (0.07)	0.83 (0.10)	0.66 (0.11)	0.54 (0.19)	0.54 (0.09)
	2000	0.96 (0.10)	0.89 (0.20)	0.89 (0.09)	0.71 (0.11)	0.54 (0.23)	0.54 (0.07)
	2500	0.98 (0.08)	0.91 (0.17)	0.87 (0.10)	0.69 (0.17)	0.65 (0.25)	0.52 (0.09)
0.70	1500	0.92 (0.12)	0.80 (0.20)	0.76 (0.12)	0.60 (0.13)	0.56 (0.18)	0.53 (0.08)
	2000	0.93 (0.12)	0.80 (0.24)	0.83 (0.11)	0.63 (0.15)	0.57 (0.22)	0.53 (0.08)
	2500	0.91 (0.16)	0.74 (0.31)	0.80 (0.10)	0.60 (0.20)	0.64 (0.24)	0.51 (0.08)
0.80	1500	0.80 (0.14)	0.62 (0.23)	0.71 (0.13)	0.53 (0.12)	0.53 (0.19)	0.52 (0.08)
	2000	0.79 (0.15)	0.55 (0.25)	0.76 (0.13)	0.55 (0.15)	0.48 (0.23)	0.52 (0.08)
	2500	0.77 (0.13)	0.45 (0.27)	0.75 (0.10)	0.46 (0.16)	0.64 (0.24)	0.51 (0.08)
0.90	1500	0.69 (0.11)	0.46 (0.15)	0.67 (0.10)	0.50 (0.11)	0.51 (0.19)	0.51 (0.08)
	2000	0.70 (0.09)	0.37 (0.15)	0.72 (0.10)	0.46 (0.13)	0.51 (0.21)	0.53 (0.10)
	2500	0.72 (0.08)	0.37 (0.16)	0.71 (0.09)	0.38 (0.13)	0.60 (0.25)	0.51 (0.08)

obtained to check if error occurred more often in one of the two dimensions. A critical value of 5% was again used. The classification accuracy results for the dimension one items were consistently higher than those for the dimension two items with mean differences of 6% to 35% for the approximate simple and complex 40% conditions. For the complex 80% conditions, the classification accuracy results for the dimension one items were consistently higher than those for the dimension two items with mean differences of 9% to 13% for the sample size of 2500, but not for the two lower sample sizes.

Classification accuracy (complex structure items). The classification accuracy results for the subset of complex structure items are presented in Table 16. The

Table 16

Classification Accuracy for Complex Structure Items for Simulated Conditions with the SAT Parameters

		Complex Structure	
Correlation	Sample	40% (16 items)	80% (32 items)
0.60	1500	0.59 (0.11)	0.51 (0.12)
	2000	0.66 (0.10)	0.51 (0.14)
	2500	0.62 (0.12)	0.56 (0.15)
0.70	1500	0.57 (0.12)	0.52 (0.12)
	2000	0.63 (0.11)	0.52 (0.14)
	2500	0.60 (0.10)	0.56 (0.13)
0.80	1500	0.56 (0.12)	0.50 (0.12)
	2000	0.60 (0.12)	0.47 (0.14)
	2500	0.56 (0.11)	0.56 (0.14)
0.90	1500	0.55 (0.12)	0.48 (0.13)
	2000	0.58 (0.12)	0.49 (0.14)
	2500	0.53 (0.10)	0.54 (0.14)

classification accuracy results for these items were lower than the overall classification accuracy results (Table 14) with mean differences greater than or equal 5% (from 5% to 16%) for all except four cases for the complex 40% conditions (r = 0.80, n = 2500; r = 0.90, n = 1500, 2000, 2500). For the complex 80% conditions, the classification accuracy for the complex items was comparable to the overall classification accuracy with mean differences less than 5%. The complex structure items measured a composite of dimensions one and two, which makes it difficult for DETECT to classify them correctly. This likely led to the lower classification accuracy results for the complex structure items for the complex 40% conditions. However, for the complex 80% conditions, a large proportion of items (32 of 40) were complex structure items, making the classification accuracy results for the subsets of complex item comparable to those for the overall datasets.

Classification consistency and misclassification error (overall). The classification consistency and misclassification error rates were calculated for the overall tests, the dimension one and dimension two items separately, and the complex structure items. The same evaluation criterion of 85% was used for evaluating classification consistency, and a criterion of 5% was used for evaluating misclassification error.

Table 17 includes the overall classification consistency and misclassification error results. Classification consistency was above criterion for the three cases in the approximate simple conditions when the correlation was 0.60, but below criterion for the remaining correlations. For both complex conditions, classification consistency was below criterion (48% to 78%) across all conditions.

As the degree of complexity increased, classification consistency decreased consistently with mean difference greater than or equal to 5% (5% to 39%) for all except three cases (r = 0.80, n = 1500, 2500, simple vs. complex 40%; r = 0.90, n = 1500, simple vs. complex 40%). Thus, degree of complexity showed a negative impact on classification consistency.

Table 17

Classification Consistency and Misclassification Error Rates between Primary and Cross-Validation Samples for Simulated Conditions with the SAT Parameters

		Structure					
		Simple		Complex 40%		Complex 80%	
Correlation	Sample	CC	ME	CC	ME	CC	ME
0.60	1500	0.92 (0.09)	0.01 (0.02)	0.73 (0.12)	0.11 (0.05)	0.50 (0.14)	0.21 (0.09)
	2000	0.88 (0.18)	0.02 (0.05)	0.78 (0.10)	0.09 (0.05)	0.50 (0.16)	0.21 (0.11)
	2500	0.91 (0.16)	0.02 (0.06)	0.78 (0.13)	0.10 (0.06)	0.59 (0.23)	0.20 (0.10)
0.70	1500	0.80 (0.16)	0.05 (0.06)	0.65 (0.12)	0.14 (0.06)	0.54 (0.12)	0.22 (0.10)
	2000	0.79 (0.21)	0.04 (0.09)	0.70 (0.13)	0.12 (0.07)	0.52 (0.16)	0.21 (0.10)
	2500	0.77 (0.23)	0.06 (0.12)	0.70 (0.14)	0.15 (0.09)	0.56 (0.24)	0.20 (0.11)
0.80	1500	0.64 (0.14)	0.11 (0.10)	0.61 (0.12)	0.19 (0.08)	0.48 (0.13)	0.21 (0.10)
	2000	0.67 (0.16)	0.15 (0.15)	0.62 (0.13)	0.15 (0.07)	0.53 (0.15)	0.25 (0.12)
	2500	0.71 (0.16)	0.25 (0.18)	0.67 (0.16)	0.22 (0.10)	0.52 (0.24)	0.19 (0.09)
0.90	1500	0.59 (0.13)	0.22 (0.10)	0.58 (0.14)	0.21 (0.08)	0.53 (0.12)	0.25 (0.10)
	2000	0.67 (0.16)	0.29 (0.13)	0.59 (0.15)	0.20 (0.09)	0.52 (0.16)	0.24 (0.12)
	2500	0.73 (0.13)	0.34 (0.12)	0.63 (0.18)	0.26 (0.11)	0.53 (0.21)	0.21 (0.12)

Note. CC is classification consistency, and ME is misclassification error.

As the correlation between dimensions increased, classification consistency decreased consistently with mean differences greater than or equal to 5% (5% to 16%) for

11 of 18 cases for the approximate simple and complex 40% conditions. Six of these 11 cases occurred for the approximate simple structure conditions when the correlation went from 0.60 to 0.70 and from 0.70 to 0.80. Three of these 11 cases occurred for the complex 40% conditions when correlation went from 0.60 to 0.70. For the complex 80% conditions, inconsistent mean differences were found with one significant negative mean difference of 5% (n = 1500, 0.80 vs. 0.90) and one significant positive mean difference of 6% (n = 1500, 0.7 vs. 0.80). Thus, an interaction effect was found between correlation and degree of complexity: correlation between dimensions showed a negative impact on classification consistency for the approximate simple structure conditions when the correlation went from 0.6 to 0.80, a negative impact on classification consistency for the complex 40% conditions when correlation went from 0.60 to 0.70, and no impact on classification consistency for the complex 80% conditions. Again, for the complex 80% conditions, classification consistency clustered within a narrow range of 48% to 59%. This narrow range led to the insignificant impact for the complex 40% conditions and the interaction effect between correlation and degree of complexity.

As sample size increased, classification consistency decreased with mean differences greater than or equal to 5% (from 5% to 9%) for seven of 24 cases (simple, r = 0.90, 1500 vs. 2000, 2000 vs. 2500; complex 40%, r = 0.60, 0.70, 1500 vs. 2000, r = 0.80, 2000 vs. 2500; complex 80%, r = 0.60, 2000 vs. 2500, r = 0.80, 1500 vs. 2000). These seven cases occurred without an identifiable pattern, thus sample size did not show an impact on classification consistency.

The misclassification error rate was satisfactory (less than or equal to 5%) only for the approximate simple condition when the correlation between dimensions was 0.60

or 0.70 (for the correlation 0.70 conditions, the sample size was 1500 or 2000). As the degree of complexity increased, misclassification error increased consistently with mean differences greater than or equal to 5% (from 5% to 12%) for all cases for the correlations of 0.60 and 0.70. For the correlations of 0.80 and 0.90, inconsistent mean differences were found with two significant positive mean differences of 8% to 10% for the correlation of 0.80 (n = 1500, simple vs. complex 40%; n = 2000, complex 40% vs. complex 80%) and three significant negative mean differences of 5% to 9% for the correlation of 0.90 (n = 2000, 2500, simple vs. complex 40%; n = 2500, complex 40% vs. complex 80%). Therefore, there was an interaction effect between degree of complexity and correlation. Degree of complexity showed a consistent positive impact on misclassification error for the correlations of 0.60 and 0.70. Degree of complexity did not show a consistent impact on misclassification error for the correlations of 0.80 and 0.90.

As the correlation between dimensions increased, misclassification error increased consistently with mean differences greater than or equal to 5% for nine of 18 cases for the approximate simple and complex 40% conditions. Five of these nine cases occurred for the approximate simple structure conditions when the correlation went from 0.80 to 0.90. The other four cases spread out in the complex 40% conditions without an identifiable pattern (n =1500, 0.70 vs. 0.80; n = 2000, 0.80 vs. 0.90; n = 2500, 0.60 vs. 0.70, and 0.70 vs. 0.80). For the complex 80% conditions, the mean differences were all less than 5%. Therefore, there was an interaction effect between correlation and degree of complexity. Correlation between dimensions showed a positive impact on misclassification error for the approximate simple structure conditions when the

correlation went from 0.80 to 0.90. Correlation between dimensions did not show an impact on misclassification error for both of the complex structure conditions.

As sample size increased, inconsistent results were found with five significant positive mean differences of 5% to 10% (simple, $r = 0.80$, 2000 vs. 2500, $r = 0.90$, 1500 vs. 2000, 2000 vs. 2500; complex 40%, $r = 0.80$, 2000 vs. 2500, $r = 0.90$, 2000 vs. 2500) and one negative mean difference of 6% (complex 80%, $r = 0.80$, 2000 vs. 2500) in misclassification error. Thus, sample size did not show a consistent impact on misclassification error.

Classification consistency and misclassification error (dimensions one and two items and complex items). Table 18 contains the classification consistency results for the dimension one and dimension two items separately. Table 19 contains the classification consistency results for the subsets of complex structure items. The same critical value of 5% was used to evaluate mean differences in classification consistency and misclassification error between the dimensions. The classification consistency results for the dimension one items were relatively higher than those for the dimension two items for the approximate simple and complex 40% conditions with mean differences of 6% to 15% for all except the three complex 40% conditions ($r = 0.60$, $n = 1500$; $r = 0.90$, $n =$ 1500, 2500). Mean differences for the complex 80% conditions were all less than 5%.

The misclassification error rates were lower for the dimension one items than those for the dimension two items for the approximate simple and complex 40% conditions with mean differences of 6% to 32% for all except three cases for the approximate simple structure conditions ($r = 0.60$, $n = 1500$, 2000, and 2500). For the complex 80% conditions, for the sample size of 2500, misclassification error was

Table 18

Classification Consistency and Misclassification Error Rates for Dimensions One and Two for the SAT Parameters

		Structure											
		Simple				Complex 40%				Complex 80%			
		Dim. I		Dim. II		Dim. I		Dim. II		Dim. I		Dim. II	
Correlation	Sample	CC	ME	CC	ME	CC	ME	CC	ME	CC	ME	CC	ME
0.60	1500	0.97 (0.08)	0.00 (0.01)	0.88 (0.12)	0.03 (0.03)	0.75 (0.14)	0.05 (0.05)	0.71 (0.13)	0.18 (0.07)	0.51 (0.17)	0.20 (0.14)	0.50 (0.15)	0.22 (0.09)
	2000	0.93 (0.13)	0.00 (0.03)	0.83 (0.23)	0.03 (0.08)	0.82 (0.12)	0.03 (0.03)	0.73 (0.12)	0.15 (0.08)	0.50 (0.19)	0.20 (0.16)	0.51 (0.17)	0.22 (0.10)
	2500	0.95 (0.11)	0.00 (0.04)	0.87 (0.21)	0.04 (0.08)	0.84 (0.13)	0.05 (0.04)	0.72 (0.16)	0.16 (0.09)	0.61 (0.25)	0.13 (0.15)	0.57 (0.23)	0.27 (0.14)
0.70	1500	0.86 (0.14)	0.02 (0.05)	0.74 (0.19)	0.08 (0.09)	0.68 (0.13)	0.09 (0.07)	0.62 (0.14)	0.20 (0.10)	0.53 (0.14)	0.21 (0.15)	0.55 (0.14)	0.24 (0.09)
	2000	0.85 (0.19)	0.02 (0.05)	0.73 (0.25)	0.07 (0.13)	0.74 (0.15)	0.05 (0.05)	0.66 (0.14)	0.20 (0.10)	0.52 (0.20)	0.19 (0.17)	0.53 (0.15)	0.24 (0.10)
	2500	0.84 (0.16)	0.02 (0.07)	0.69 (0.32)	0.10 (0.16)	0.75 (0.15)	0.07 (0.06)	0.65 (0.16)	0.22 (0.15)	0.58 (0.26)	0.14 (0.16)	0.54 (0.24)	0.27 (0.15)
0.80	1500	0.70 (0.15)	0.06 (0.07)	0.58 (0.18)	0.16 (0.14)	0.64 (0.14)	0.12 (0.09)	0.58 (0.15)	0.25 (0.11)	0.49 (0.16)	0.22 (0.15)	0.47 (0.14)	0.21 (0.09)
	2000	0.74 (0.17)	0.08 (0.10)	0.60 (0.20)	0.23 (0.21)	0.66 (0.16)	0.07 (0.06)	0.58 (0.14)	0.22 (0.12)	0.53 (0.19)	0.26 (0.20)	0.54 (0.15)	0.25 (0.10)
	2500	0.77 (0.12)	0.12 (0.11)	0.65 (0.23)	0.37 (0.25)	0.72 (0.18)	0.11 (0.07)	0.63 (0.16)	0.32 (0.16)	0.52 (0.27)	0.12 (0.12)	0.52 (0.23)	0.26 (0.14)
0.90	1500	0.64 (0.15)	0.13 (0.07)	0.55 (0.15)	0.30 (0.15)	0.60 (0.16)	0.14 (0.09)	0.56 (0.16)	0.27 (0.12)	0.53 (0.14)	0.25 (0.16)	0.52 (0.15)	0.25 (0.09)
	2000	0.71 (0.17)	0.16 (0.09)	0.63 (0.17)	0.43 (0.19)	0.63 (0.17)	0.11 (0.06)	0.56 (0.16)	0.30 (0.14)	0.52 (0.20)	0.25 (0.20)	0.52 (0.16)	0.24 (0.11)
	2500	0.77 (0.14)	0.18 (0.08)	0.69 (0.15)	0.50 (0.17)	0.65 (0.20)	0.13 (0.08)	0.61 (0.19)	0.38 (0.18)	0.54 (0.25)	0.17 (0.18)	0.53 (0.21)	0.26 (0.13)

Note. CC is classification consistency, and ME is misclassification error.

Table 19

Classification Consistency and Misclassification Error Rates for Complex Structure Items for Simulated Conditions with the SAT Parameters

		Complex Structure			
		40% (16 items)		80% (32 items)	
Correlation	Sample	CC	ME	CC	ME
0.60	1500	0.65 (0.14)	0.23 (0.10)	0.49 (0.15)	0.23 (0.11)
	2000	0.68 (0.14)	0.19 (0.09)	0.48 (0.18)	0.24 (0.13)
	2500	0.69 (0.14)	0.21 (0.10)	0.58 (0.25)	0.22 (0.12)
0.70	1500	0.61 (0.14)	0.24 (0.10)	0.52 (0.13)	0.25 (0.11)
	2000	0.67 (0.16)	0.22 (0.11)	0.51 (0.18)	0.23 (0.12)
	2500	0.66 (0.15)	0.23 (0.11)	0.56 (0.26)	0.21 (0.13)
0.80	1500	0.59 (0.15)	0.25 (0.12)	0.47 (0.15)	0.24 (0.12)
	2000	0.59 (0.17)	0.21 (0.11)	0.52 (0.17)	0.28 (0.14)
	2500	0.67 (0.17)	0.28 (0.11)	0.51 (0.25)	0.20 (0.10)
0.90	1500	0.57 (0.18)	0.25 (0.13)	0.52 (0.13)	0.27 (0.12)
	2000	0.60 (0.19)	0.23 (0.12)	0.51 (0.17)	0.27 (0.14)
	2500	0.62 (0.20)	0.28 (0.12)	0.53 (0.23)	0.23 (0.13)

Note. CC is classification consistency, and ME is misclassification error.

consistently lower across the four correlation conditions for the dimension one items than the dimension two items with mean difference of 9% to 14%.

As seen from Table 19, classification consistency results for the complex structure items were lower than the overall classification accuracy results (Table 17) with mean differences greater than or equal 5% (from 5% to 16%) only for three cases for the approximate simple structure conditions ($r = 0.60$, $n = 1500, 2000, 2500$). The classification consistency results for the complex structure items were comparable to those for all items.

The misclassification error rates for the complex structure items were consistently lower than the misclassification error rates for all items with mean differences from 6% to 12% for the complex 40% conditions with a correlation of 0.80 and lower. All mean differences were less than 5% for the complex 80% conditions. Thus the misclassification error rates for the complex structure items were comparable to the misclassification error rates for all items for the complex 80% conditions.

LSAT Results versus SAT Results

When the D_{max}, r_{max}, classification accuracy, classification consistency, and misclassification error results for the LSAT parameters were compared with those for the SAT parameters, the LSAT results were generally better than the SAT results. For the D_{max} index, the LSAT results were consistently higher than the SAT results with mean differences greater than or equal to 0.05 for all the approximate simple structure conditions and for the correlations of 0.60, 0.70, and 0.80 for the complex 40% conditions (see Tables 6 and 13). The D_{max} values were comparable for remaining complex 40% conditions and the complex 80% conditions.

For the r_{max} index, the LSAT results were consistently higher than the SAT results with mean differences greater than or equal to 0.05 for all except one case (complex 40%, $r = 0.90$, $n = 1500$) for the approximate simple and complex 40% conditions (see Tables 6 and 13). For the complex 80% conditions, the LSAT results were consistently higher than the SAT results for the correlations of 0.60 and 0.70. For the correlations of 0.80 and 0.90, the results were comparable.

For the classification accuracy, the LSAT results were consistently higher than the SAT results with mean differences greater than or equal to 5% except for one case

(complex 80%, $r = 0.90$, $n = 2500$; see Tables 7 and 14). For the classification consistency, the LSAT results were consistently higher than the SAT results with mean differences greater than or equal to 5% except for two cases (complex 80%, $r = 0.90$, $n = 1500, 2000$; see Tables 10 and 17). For the misclassification error rate, the LSAT results were consistently lower than the SAT results with mean differences greater than or equal to 5% except for eight of 36 cases. For the approximate simple structure with a correlation of 0.60, misclassification error rates for both were lower than 5% for the three sample size conditions. Misclassification error rates were comparable in this case. All other cases with mean differences less than 5% (simple, $r = 0.70$, $n = 2000$; complex 80%, $r = 0.80$, $n = 1500, 2500$, $r = 0.90$, $n = 1500, 2500$) spread across conditions without an identifiable pattern. Thus, the misclassification error rates for the LSAT parameters were consistently lower than those for the SAT parameters for the approximate simple structure conditions with correlations of 0.70 to 0.90 and for the complex 40% and complex 80% conditions.

Since the discrimination parameters of items are different for the two sets of parameters, differences found between the LSAT and the SAT results could be attributable to differences in item discrimination parameters. The LSAT parameters included moderate to high discriminating items while the SAT parameters included low to high discriminating items. The existence of low discriminating items (20% of total items) in the SAT parameters could be responsible for the degraded SAT results.

Another possible explanation has to do with the difference in angular departures between the two dimensions. For the LSAT parameters, the dimension one items had angular directions from 0° to 15°, and the dimension two items had angular directions

from 75° to 90°. The angular difference between the two dimensions was 60°. For the SAT parameters, the dimension one items had angular directions from 0° to 10°, and the dimension two items had angular directions from 30° to 55°. The angular difference was 20°. When the angular departure between the two dimensions becomes smaller, it is more difficult for DETECT to distinguish between the two dimensions. This could be another reason why the LSAT results were much better than the SAT results.

Regression Analysis for D_{max} and Classification Accuracy

The D_{max} and classification accuracy results in Tables 6 and 7 and Tables 13 and 14 indicated that higher D_{max} values were associated with higher classification accuracy results. An inspection of these tables suggested that all conditions with a D_{max} value of greater than or equal to 0.15 obtained classification accuracy greater than or equal to 85%, the criterion used for evaluating classification accuracy. When at least 85% of the items are accurately classified, we should be relatively confident in considering a test as multidimensional. Thus, a classification accuracy of 85% was used as an indicator of multidimensionality in a test. A regression analysis was conducted in which D_{max} was regressed on classification accuracy to confirm empirically this finding by comparing the predicted D_{max} with a classification accuracy of 85% and the 0.15 observational finding.

First, the scatter plot with D_{max} on the x axis and classification accuracy on the y axis was examined to see if the relationship between the two variables was linear. As shown in Figure 11, the scatter plot showed a curvilinear relationship between the two variables. Different curvilinear models as well as the linear model were fit to the data. The regression functions for the different models considered are presented in Table 20.

Figure 11 is the scatter plot with the four corresponding reference lines. Based on Table 20, all models exhibited adequate fit with significant *F* tests at the alpha level of 0.05. The increment in *R* squared can be tested using the following formula (Pedhazur, 1982):

$$F = \frac{(R^2_{k_1} - R^2_{k_2})/(k_1 - k_2)}{(1 - R^2_{k_1})/(N - k_1 - 1)},$$

where *F* is the *F* test with degree of freedom $((k_1 - k_2),(N - k_1 - 1))$, $R^2_{k_1}$ is the *R* squared associated with the model to be tested, $R^2_{k_2}$ is the *R* squared associated with the baseline model, k_1 is the degree of freedom associated with the model to be tested, k_2 is the degree of freedom associated with the baseline model, and *N* is the sample size.

Table 20

Regression Analysis Results for D_{max} and Classification Accuracy

	Model Summary					Parameter Estimates			
	R^2	*F*	*df1*	*df2*	Sig.	B_0	B_1	B_2	B_3
Linear	0.41	4944.46	1	7198	0.00	-0.16	0.43		
Quadratic	0.56	4596.69	2	7197	0.00	0.63	-1.92	1.61	
Cubic	0.59	5087.65	2	7197	0.00	0.25	0.00	-1.42	1.50
Exponential	0.59	10163.07	1	7198	0.00	0.02	2.30		

However, this test could not be used to test whether the increment in *R* squared was significant in the present study because the linear and exponential models and the quadratic and cubic models had the same degrees of freedom. Besides, the sample size of 7200 would likely make any increment statistically significant if the test could be conducted. In order to determine the best model, a critical value of 0.05 was used to assess increment in *R* squared, which meant an increment of 5% variance explained indicated a significant difference between models.

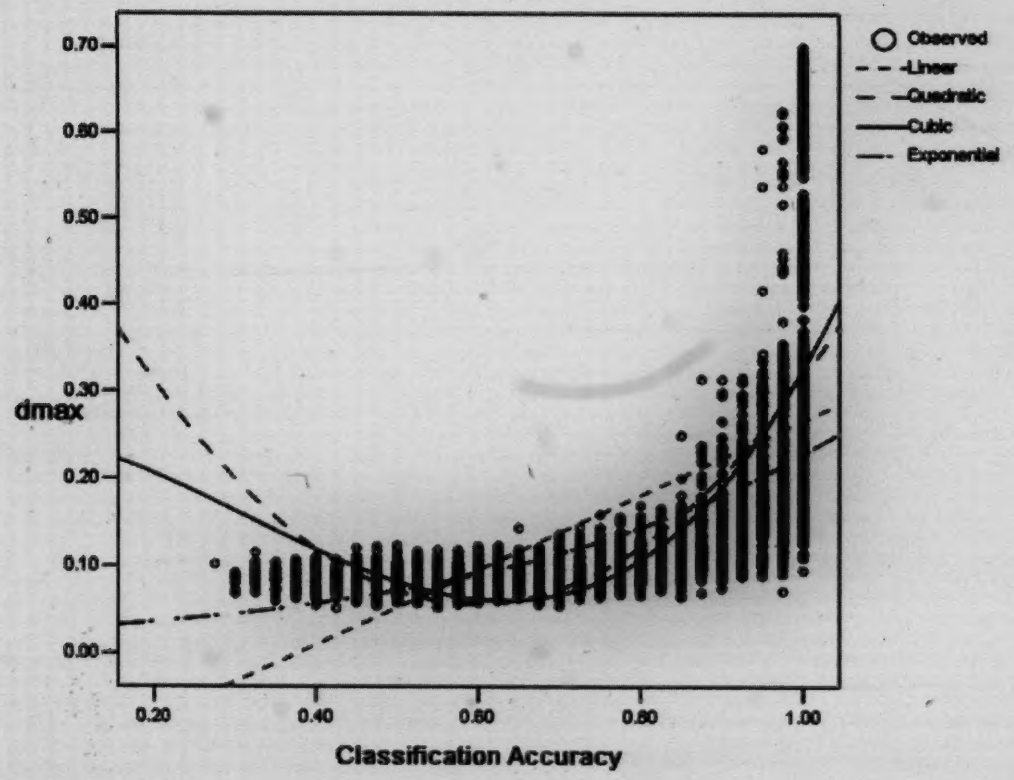


Figure 11. Linear and nonlinear fitting functions for regressing D_{max} on classification accuracy.

The increment between the exponential and the linear models was 0.18, so the exponential model explained significantly more variance. The increment between the quadratic model and the exponential model was -0.03, and the increment between the cubic model and the exponential model was 0.00. Thus, these three models provided comparable fit. Examination of the reference lines in Figure 11 suggested that these three models provided adequate fit to the test data except at the end of the scale corresponding to the 100% classification accuracy. However, since the 100% classification accuracy was not used for prediction, this proved not to be a problem.

A classification accuracy of 85% was used as the cut-score to predict the critical value for D_{max}. Using the regression functions obtained for the exponential, quadratic,

and cubic models, predicted D_{max} values of 0.16, 0.16, and 0.15, respectively, were obtained. These values are close to the 0.15 observational value. Thus, a critical value of 0.15 was proposed for evaluating D_{max}. A D_{max} value equal to or greater than 0.15 suggests moderate to strong multidimensionality, while values less than 0.15 suggest essential unidimensionality to weak multidimensionality.

Regression Analysis for r_{max} and Classification Accuracy

r_{max} was also closely related to classification accuracy. An inspection of Tables 6, 7, 13, and 14 suggested that higher r_{max} values are associated with higher classification accuracy results. However, what value should the minimum classification accuracy be to indicate simple or approximate simple structure? Results in the simulation study revealed that even for some complex structure conditions the classification accuracy could exceed 90% (e.g., for the complex 40% and correlation 0.8 condition in the LSAT results, classification accuracy was 92% when sample size was 2000). Thus, a high classification accuracy of 95% was used as an indicator of simple or approximate simple structure, which will reduce the chance of setting a critical value that is overly liberal. A visual inspection of the simulation results suggested that all conditions with r_{max} values greater than or equal to 0.60 obtained classification accuracy greater than or equal to 95%. Thus, a regression analysis was conducted in which r_{max} was regressed on classification accuracy to confirm the visual inspection with statistical outcome and to obtain the predicted r_{max} with a classification accuracy of 95%.

First, the scatter plot with r_{max} as the *x* axis and classification accuracy as the *y* axis was examined to see if the relationship between the two was linear (Figure 12). The

scatter plot showed a curvilinear relationship. Different models including linear, exponential, quadratic, and cubic were fitted. The regression functions are presented in Table 21. Figure 12 is the scatter plot with the four corresponding reference lines. Based

Table 21

Regression Analysis Results for r_{max} and Classification Accuracy

	Model Summary					Parameter Estimates			
	R^2	F	$df1$	$df2$	Sig.	B_0	B_1	B_2	B_3
Linear	0.59	10370.11	1	7198	0.00	-0.16	0.80		
Quadratic	0.76	11147.18	2	7197	0.00	1.11	-2.98	2.58	
Cubic	0.78	12422.59	2	7197	0.00	0.51	0.00	-2.06	2.27
Exponential	0.69	15839.08	1	7198	0.00	0.11	1.68		

on Table 21, all models exhibited adequate fit given that they all had significant F tests at the alpha level of 0.05. A critical value of 0.05 was again used to assess increment in R squared to determine the best model for prediction. The increment between the exponential and the linear models was 0.10, so the exponential model explained significantly more variance. The increment between the quadratic and the exponential models was 0.07, so the quadratic model explained significantly more variance. The increment between the cubic and the quadratic models was only 0.02. Thus, the quadratic and the cubic models provided comparable fit. Examination of the reference lines in Figure 12 also suggested that these two models provided adequate fit to the test data.

A classification accuracy of 95% was used to predict the critical value for r_{max}. Using the regression functions obtained for the quadratic and the cubic models, predicted r_{max} values of 0.61 and 0.60 were obtained, which are close to the 0.60 observational value. Thus, a critical value of 0.60 was proposed for evaluating r_{max}. Values greater than

or equal to 0.60 suggest simple or approximate simple structure, while values less than 0.60 suggest complex structure.

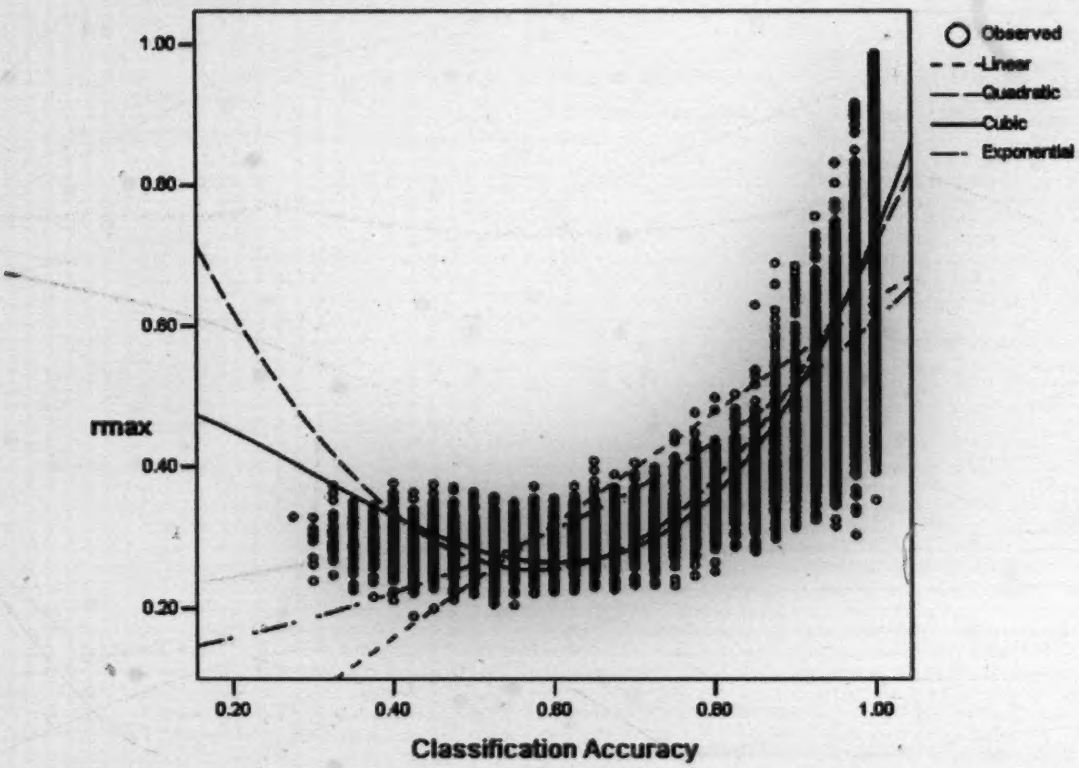


Figure 12. Linear and nonlinear fitting functions for regressing r_{max} on classification accuracy.

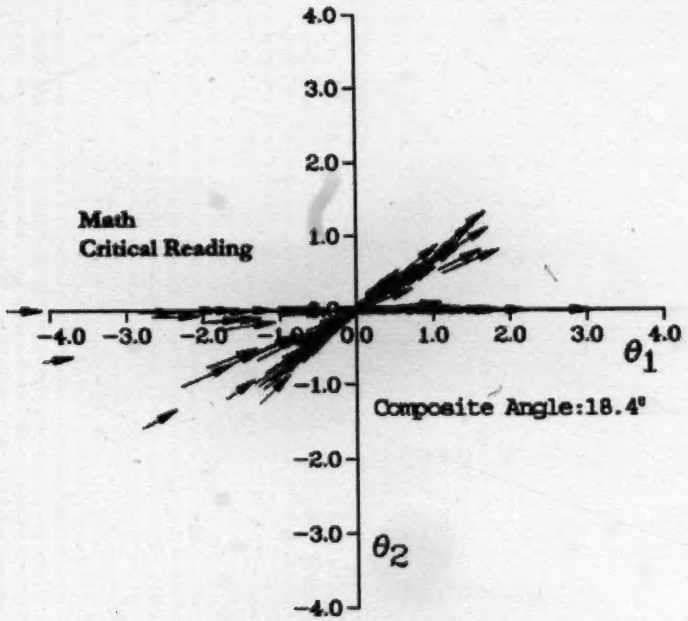
Real Data Studies

Data from the 2005 March administration of the SAT were analyzed to establish the connection between the simulation results and real testing outcomes. Three analyses were conducted, one at the composite test level and two at the Math subtest level.

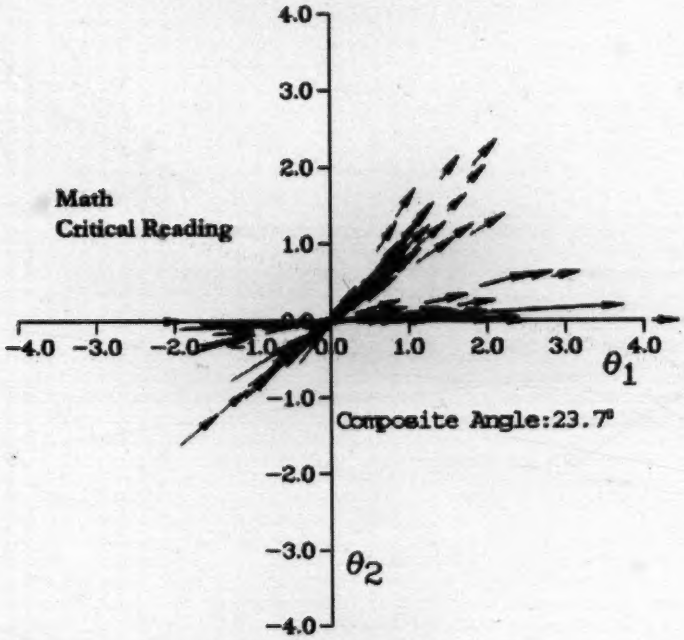
At the composite test level, the dataset contained Math and Critical Reading items. The hypothesized dimensionality was two-dimensional with these two content categories as the two dimensions. The NOHARM analysis revealed that the correlation between the two dimensions was 0.72. The item parameters obtained from the NOHARM analysis

were used to produce the vector plot of items for the SAT 2005 March composite test (see top graph of Figure 13). The graph clearly indicated an approximate simple structure with moderate correlation between dimensions. The corresponding condition in the simulation study is the approximate simple structure condition with a correlation of 0.70 and a sample size of 2500 for the data simulated with the SAT parameters.

The D_{max} and r_{max} indices obtained from DETECT were 0.43 and 0.86, respectively, indicating weak multidimensionality with approximate simple structure using Kim's (1994) evaluation criteria. The classification accuracy and consistency were both 99%, and the misclassification error rate was 0.00. The classification results for the composite test were higher than those obtained from the corresponding simulated condition (classification accuracy 99% vs. 82% and classification consistency 0.99 vs. 0.77, see Tables 14 and 17). It should be noted that the real test consisted of 121 items with a variety of combinations of a_1 and a_2 parameters. A large number of items had extremely high discrimination parameters on one dimension and extremely low discrimination parameters on the other dimension (e.g., an item with a_1 of 1.34 and a_2 of 0.04). In contrast, although the item parameter pairs used for simulation approximated the distribution of the item parameters of the SAT 2003 field trial data, they did not contain extreme pattern pairs. Furthermore, when the item parameters of the SAT 2003 field trial data were compared to those of the SAT 2005 March administration data, it was found that the SAT 2005 March administration data had a clearer dimensional structure. As seen from the vector plot of items for the SAT 2003 field trial (see bottom graph of



(1) SAT 2005 March administration



(2) SAT 2003 field trial

Figure 13. Vector plots of items for the composite tests.

Figure 13), the items on the field trial have a wider span than the items on the 2005 March administration. These differences likely produced the discrepancy between the real and the simulated conditions given the item parameters used for the simulation were adopted from the 2003 field trial item parameters.

For the Math subtest, two analyses were conducted. For the first dataset, skill 1 and skill 4 items were analyzed together. The hypothesized dimensionality was two-dimensional with these two skills as the dimensions. This dataset was expected to exhibit a low degree of complexity that corresponded to the simulated complex 40% conditions. A value of 0.57 was obtained for the correlation between dimensions. The item parameters obtained from NOHARM were used to produce the vector plot of items (top graph of Figure 14). As seen from this graph, most of the skill 1 and 4 items have a clear separation. The two clusters of items do not have a large separation angle, but item overlap is minimal. The corresponding condition in the simulation study is the complex 40% condition with a correlation of 0.60 and a sample size of 2500.

The D_{max} and r_{max} indices obtained from DETECT were 0.13 and 0.51, respectively, indicating weak multidimensionality with complex structure. The classification accuracy and consistency were 83% and 74%, respectively, and the misclassification error rate was 4%. Compared to the results in the corresponding simulated condition (78% and 78% respectively), the results were close between the simulated and the real data conditions.

For the second dataset, skills 2 and 3 items were analyzed together. The hypothesized dimensionality was again two-dimensional with these two skills as dimensions. However, in contrast to the previous dataset, this dataset was expected to

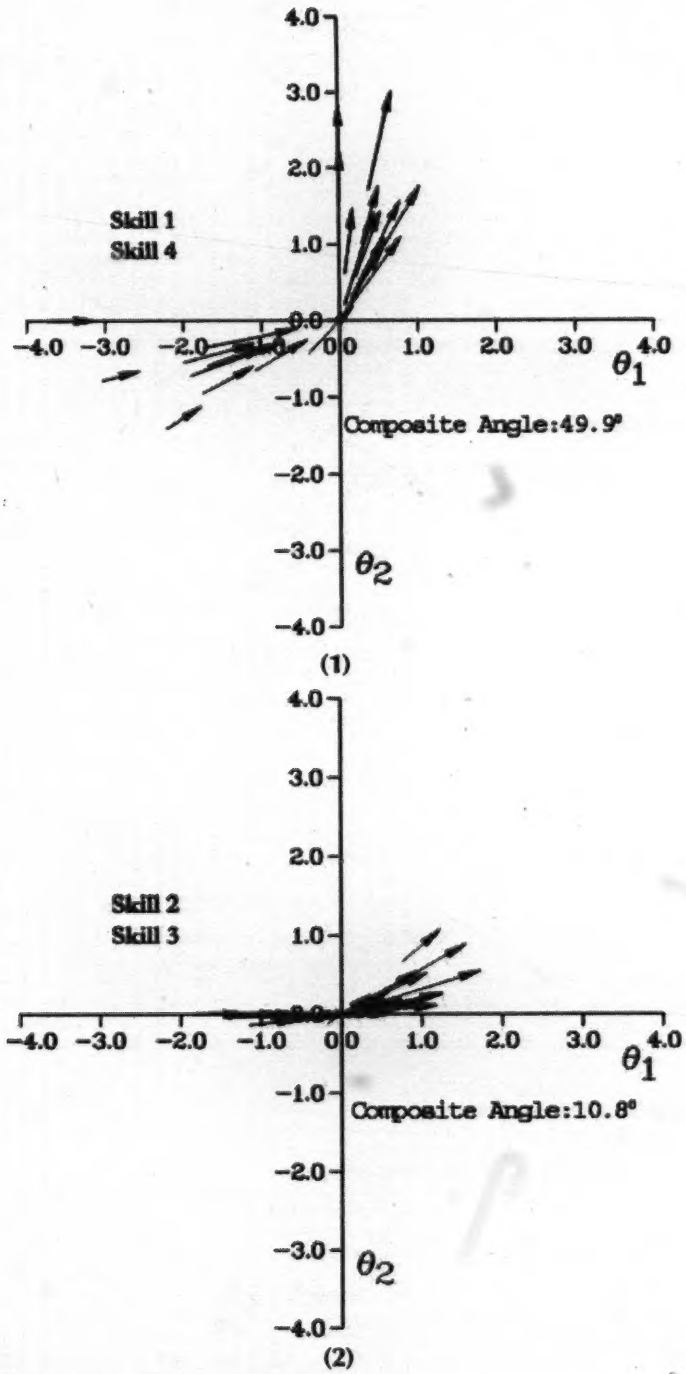


Figure 14. Vector plots of items for the Math subtest.

exhibit a high degree of complexity that corresponded to the simulated complex 80% conditions. A value of 0.80 was obtained for the correlation between dimensions. The item parameters obtained from NOHARM were used to produce the vector plot of items (bottom graph of Figure 14). As seen from the graph, most of the skill 2 and skill 3 items clustered together indicating a complex dimensional structure. The corresponding condition in the simulation study is the complex 80% condition with a correlation of 0.80 and a sample size of 2500.

The D_{max} and r_{max} indices obtained from DETECT were 0.13 and 0.42, respectively, indicating weak multidimensionality with complex structure. The classification accuracy and consistency were 61% and 48% respectively, and the misclassification error rate was 35%. Compared to the results in the corresponding simulated condition, the classification accuracy and consistency results were comparable (61% vs. 58% and 48% vs. 52% respectively). The two subtest level results showed strong correspondence with the results from the associated simulated conditions.

Chapter 6: Discussion and Conclusions

Five sections are included in this chapter. First, the research questions are revisited together with a brief summary of the methods used for the present study. Second, a summary and discussion of the results are provided. Third, the conclusions from the present study are presented. Fourth, the limitations of the study are outlined. Fifth, the educational and practical implications from the study and recommendations for future research are discussed.

Restatement of Research Questions and Summary of Methods

The purpose of this study was to evaluate the performance of DETECT under conditions of both approximate simple and complex structures. The accuracy and consistency with which DETECT can classify dichotomously scored items into dimensions is key to helping researchers and practitioners identify the dimensional structure of a test. DETECT, as shown in literature, produces accurate and consistent classification results when the data possess simple or approximate simple structure (Roussos & Ozbek, 2003; Stout et al., 1996; Zhang & Stout, 1999b). However, when data with complex structure are analyzed, DETECT has been shown to perform inconsistently across samples in simulated and real data analyses (Gierl et al., 2005; Leighton et al., in press; Uribe-Zarain, Nandakumar, & Yu, 2005). In the present study, the performance of DETECT was evaluated for data with complex structure through simulation and with incorporation of data with the approximate simple structure to serve as the baseline for comparison. The impact of three factors on D_{max}, r_{max}, classification accuracy, and classification consistency was studied systematically through simulation. Real data studies using DETECT to analyze data with hypothesized dimensional structure were

also conducted to bring a sense of reality into the simulation results. The research questions addressed in this study included:

1. Are the D_{max} and r_{max} indices, classification accuracy, and classification consistency of DETECT influenced by the presence of different degrees of complexity in data structure?
2. Are the D_{max} and r_{max} indices, classification accuracy, and classification consistency of DETECT influenced by the correlations among dimensions?
3. Are the D_{max} and r_{max} indices, classification accuracy, and classification consistency of DETECT influenced by the sample size?
4. Is there a relationship between the D_{max} index and classification accuracy? If there is a relationship, what is the direction of the relationships?
5. Is there a relationship between the r_{max} index and classification accuracy? If there is a relationship, what is the direction of the relationships?

To answer the first question, tests with different percentages of items measuring multiple dimensions were simulated to create different degrees of complexity in the data. Three levels were created: one for the approximate simple structure, with 0% of the items measuring multiple dimensions and two for the complex structure, with 40% and 80% items measuring multiple dimensions. To answer the second question, different values of correlation between dimensions were selected for simulation. Four levels were created with moderate to high correlations (0.60 to 0.90 with an increment of 0.10). To answer the third question, three different sample sizes (1500, 2000, and 2500) were used for simulation.

To answer the fourth question, a regression analysis was conducted setting classification accuracy as the independent variable and D_{max} as the dependent variable. After the relationship between the two variables was established, a classification accuracy value was selected to predict the critical value for D_{max} to serve as a new criterion for indicating multidimensionality.

Similarly, to answer the fifth question, a regression analysis was conducted setting classification accuracy as the independent variable and r_{max} ad the dependent variable. After the relationship between the two was established, a value was selected for classification accuracy to predict a critical value for r_{max} to serve as a new criterion for indicating the nature of the data structure—simple or complex.

Discussion of Results and Conclusions

Simulation Results

Impact of the degree of complexity in data structure. For the data simulated with the LSAT parameters, degree of complexity showed a consistent negative impact on D_{max}, r_{max}, classification accuracy, and classification consistency. However, an interaction effect was found between degree of complexity and correlation between dimensions on misclassification error. Misclassification error rate increased only when the degree of complexity increased from 40% to 80% for the correlations of 0.60 to 0.80. Misclassification error rate increased when the degree of complexity increased across all levels for the correlation of 0.90. This interaction effect was likely caused by the clustering of misclassification error rates close to the lowest possible value of 0% for the approximate simple structure conditions and for the three lower correlation conditions for the complex 40% conditions.

For the data simulated with the SAT parameters, degree of complexity did not show a significant impact on D_{max}. However, it did show an impact on r_{max} with an interaction with the correlation between dimensions. r_{max} decreased when the degree of complexity increased for the correlation of 0.60. r_{max} decreased when the degree of complexity increased from 0% to 40% for the correlation of 0.70. For other correlation conditions, degree of complexity did not show a significant impact on r_{max}. The interaction effect was again caused by the clustering of r_{max} values at the low end of the scale for the two high correlation conditions (0.80 and 0.90).

Degree of complexity also showed an impact on classification accuracy with an interaction with the correlation between dimensions. Classification accuracy decreased as the degree of complexity increased for correlations of 0.60 and 0.70. Classification accuracy decreased for the correlation of 0.80 only when the degree of complexity increased from 40% to 80%. Degree of complexity did not show a consistent impact on classification accuracy for the correlation of 0.90. The clustering of classification accuracy values at the low end of the scale for the two high correlation conditions (0.80 and 0.90) and the complex 80% conditions accounted for the interaction effect.

A consistent negative effect was found on classification consistency. Classification consistency decreased as the degree of complexity increased. Finally, there was an interaction effect between degree of complexity and correlation between dimensions on misclassification error. Misclassification error increased as the degree of complexity increased for the correlations of 0.60 and 0.70. Degree of complexity did not show a consistent impact on misclassification error for the correlations of 0.80 and 0.90.

Impact of the correlation between dimensions. For the data simulated with the LSAT parameters, correlation between dimensions showed an impact on all five statistics, D_{max}, r_{max}, classification accuracy, classification consistency, and misclassification error. However, an interaction effect was found with the degree of complexity for all five statistics. D_{max} decreased when the correlation between dimensions increased only when the approximate simple or complex 40% structures were involved. For the r_{max} index, a negative impact was found only when the approximate simple or complex 40% structures were involved. Clustering of values at the low end for the complex 80% conditions were found for both statistics. The narrow ranges resulted from this clustering effect led to the interaction effects.

For the classification accuracy, no significant impact was found for the approximate simple structure conditions. Classification accuracy decreased only when the correlation increased from 0.80 to 0.90 for the complex 40% conditions. Classification accuracy decreased when the correlation increased for the complex 80% condition. For the classification consistency, correlation did not show an impact for the approximate simple structure conditions. Classification consistency decreased only when the correlation increased from 0.70 to 0.90 for the complex 40% conditions. Classification consistency decreased when the correlation increased for the complex 80% conditions. For both classification accuracy and consistency, there was a clustering effect with values clustering at the highest possible value of 100% for the approximate simple and the low correlation (0.60) conditions for the complex 40% conditions. This led to the interaction effect between correlation and degree of complexity for both statistics.

Misclassification error rate increased only when the correlation increased from 0.80 to 0.90 for the complex 40% conditions. Misclassification error increased only when the correlation went from 0.70 to 0.80 for the complex 80% conditions.

For the data simulated with the SAT parameters, correlation between dimensions did not show a significant impact on D_{max}. However, it did show an impact on r_{max} with an interaction with the degree of complexity. r_{max} decreased only when the correlation increased from 0.60 to 0.80 for the approximate simple structure conditions. For the complex 40% and 80% conditions, correlation did not show a significant impact on r_{max}. The r_{max} values clustered at the low end of the scale for the two high correlations (0.80 and 0.90) of the complex 40% conditions and for all complex 80% conditions. The narrow ranges resulted from the clustering effect led to the interaction effect.

For the classification accuracy, classification consistency, and misclassification error, correlation between dimensions showed an impact on all three statistics with an interaction with the degree of complexity. Classification accuracy decreased as the correlation increased for the approximate simple and complex 40% conditions. Correlation between dimensions did not show consistent impact on classification accuracy for the complex 80% conditions. Classification consistency decreased when the correlation increased from 0.60 to 0.8 for the approximate simple structure conditions. Classification consistency decreased when the correlation increased from 0.60 to 0.70 for the complex 40% conditions. Correlation between dimensions did not show a consistent impact on classification consistency for the complex 80% conditions. For both classification accuracy and consistency, there was a clustering effect near the low end of

the scale for the two high correlations (0.80 and 0.90) of the complex 40% conditions and for all complex 80% conditions. This led to the interaction effects for both statistics.

Misclassification error increased when the correlation increased from 0.80 to 0.90 for the approximate simple structure conditions. For the complex 40% and 80% conditions, correlation between dimensions did not show a consistent impact on misclassification error.

Impact of the sample size. For the data simulated with the LSAT parameters, sample size did not show a significant impact on D_{max}, classification accuracy, and misclassification error. However, it did show an impact on r_{max} and classification consistency with an interaction with the degree of complexity. The r_{max} index increased as sample size increased from 1500 to 2000 for the approximate simple and complex 40% conditions. For the classification consistency, sample size did not show a significant impact on it for the approximate simple and complex 40% conditions, but for the complex 80% conditions classification consistency increased when the sample size increased from 2000 to 2500. For the data simulated with the SAT parameters, sample size did not an impact on any of the five statistics, D_{max}, r_{max}, classification accuracy, classification consistency, and misclassification error.

Discrepancies between the LSAT and SAT results. Overall, better results were obtained with the LSAT parameters than with the SAT parameters. Higher D_{max}, r_{max}, classification accuracy, and classification consistency were found together with lower misclassification error rates for the data simulated with the LSAT parameters. These discrepancies could be attributable to the inclusion of less discriminating items in the SAT parameters. Low discrimination parameters with values from 0.2 to 0.4 were

included in the SAT parameters, while the LSAT parameters only included moderate to high discrimination parameters with values from 0.5 to 1.1. This finding is consistent with research in the literature (e.g., van Abswoude et al., 2004).

Another possible explanation had to do with the difference in angular departures of the two dimensions. For the LSAT parameters, the angular departure between the two dimensions was 60°. For the SAT parameters, the angular departure between the two dimensions was 20°. When the angular departure between the two dimensions becomes smaller, it becomes more difficult for DETECT to distinguish between the two dimensions. This could be another reason why the LSAT results were much better than the SAT results.

Adequacy of classification accuracy and consistency. The successful identification of the number of dimensions underlying the responses on a test and the meaningful interpretation of the identified dimensions depend on the accuracy and consistency of the dimensionality assessment procedures. As the true underlying dimensional structure is seldom known in real testing situations, decisions on the dimensional structure of a test rely on cross validation using multiple samples. Only when consistent results are found across samples can we draw conclusions on the dimensional structure underlying a test with confidence.

Results from this study showed that DETECT worked adequately producing classification accuracy and consistency greater than or equal to the criterion of 85% for 15 of 24 conditions when data displayed approximate simple structure and for 10 of 48 conditions when data displayed complex structure. When tests with moderate to high discriminating items and clearer multidimensional structure, such as the LSAT, were

analyzed, DETECT worked well for all approximate simple structure conditions. When lower degrees of complexity (40%) were involved, DETECT produced classification accuracy and consistency greater than or equal to 85% as long as the correlation between dimensions was less than or equal to 0.80. DETECT worked satisfactorily for tests with higher degrees of complexity (80%) only when the correlation was 0.60 given a sample size of 2500.

However, when tests that included low discriminating item and possessed a messier dimensional structure, such as the SAT, were analyzed, DETECT worked satisfactorily only under limited conditions with approximate simple structure (i.e., when the correlation was 0.60). DETECT worked poorly for all complex structure conditions.

Refinement of Evaluation Criteria for D_{max} *and* r_{max}

An important issue raised in this study is the refinement of the evaluation criteria for the DETECT indices. For the D_{max} index, high classification accuracy results were obtained for many D_{max} values indicating weak multidimensionality according to Kim's (1994) criteria. For example, for the 0.60 correlation and approximate simple structure condition with the LSAT parameters, the D_{max} values were 0.61 to 0.63 indicating moderate multidimensionality. The classification accuracy obtained for these conditions was 100%. For the 0.90 correlation and approximate simple structure conditions with the LSAT parameters, the D_{max} values were 0.15 and 0.16 indicating weak multidimensionality. However, the classification accuracy remained above 90%. This outcome suggested the data were clearly two-dimensional, and the D_{max} values of 0.15 and 0.16 actually indicated, at least, moderate multidimensionality. Thus, the evaluation

criteria for the D_{max} index might be underestimating the strength of multidimensionality in a test.

A visual inspection of the simulation results for D_{max} and classification accuracy and the regression analysis between them both suggested a critical value of 0.15 when datasets with classification accuracy greater than or equal to 85% were considered multidimensional. Thus, instead of the multi-level evaluation criteria proposed by Kim (1994), a single critical value was proposed in the present study to impose a dichotomy on D_{max}. Values greater than or equal to 0.15 indicate moderate to strong multidimensionality, while values less than 0.15 indicate essential unidimensionality to weak multidimensionality.

The r_{max} index was also related to classification accuracy. For example, using the SAT parameters, when data of approximate simple structure were simulated and the correlation between dimensions was set to 0.60, the r_{max} obtained was 0.55 for the sample size of 2500. This value, according to Kim (1994), indicated complex structure. However, an analysis of the vector plot of items (Figure 8) as well as the high classification accuracy result (95%) suggested otherwise. DETECT clearly identified the items as measuring two dimensions with high accuracy and consistency. Thus, the 0.80 evaluation criterion for r_{max} might be too stringent.

A visual inspection of the simulation results for r_{max} and classification accuracy and the regression analysis between them both suggested a critical value of 0.60 when datasets with classification accuracy greater than or equal to 95% were considered as showing approximate simple to simple structure. Values greater than or equal to 0.60

indicate simple or approximate simple structure, while values less than 0.60 indicate complex structure.

Real Data Results

All three analyses conducted in the real data studies showed important consistencies with the simulation results. Analysis of the composite test obtained much higher classification accuracy and consistency (both at 99%) than the corresponding simulated condition (82% and 77%). However, as discussed in the previous chapter, the real data exhibited a cleaner dimensional structure, and this led to the discrepancy between the real and the simulated conditions. The two datasets analyzed at the Math subtest level both obtained reasonably close classification accuracy and consistency results with those obtained in the corresponding simulated conditions. The skills 1 and 4 dataset produced similar results to those obtained for the simulated complex 40% and correlation 0.60 conditions. The skills 2 and 3 dataset produced similar results to those obtained for the simulated complex 80% and correlation 0.80 conditions. The replication of the simulation results in the real data analyses illustrated the truthfulness of the simulation conditions relative to actual testing situations. The use of different item parameters that resembled real tests brought reality into the simulation study. The comparable results obtained in the simulation and real data studies also provide us with confidence in the new guidelines proposed for interpreting the D_{max} and r_{max} indices.

Conclusions

Based on the simulation and real data results, it can be concluded that DETECT can identify the dimensional structure of a test with considerable accuracy and consistency only under limited conditions when complex structure is involved. These

conditions include restrictions on the discrimination parameter of items (moderate to high), the degree of complexity involved (lower than or equal to 40%), and the correlation between dimensions (lower than or equal to 0.80). DETECT is a suitable candidate for assessing the dimensionality of a test only when approximate simple structure or low degree of complexity are involved in the data.

Degree of complexity and correlation between dimensions both have negative impacts on the DETECT indices and classification accuracy and consistency in some form, either consistent negative impact by itself or with an interaction with each other. Sample size do not have a significant or consistent impact on the DETECT indices and classification accuracy. Although it did show a positive impact on classification consistency under limited conditions, overall there is no significant and consistent impact on classification consistency.

Analyses of the DETECT indices suggest that the evaluation criteria for them are somewhat stringent. The regression analyses between the DETECT indices and classification accuracy suggest a new critical value of 0.15 for D_{max} and a new critical value of 0.60 for r_{max}.

Limitations of the Study

The present study is limited in two aspects. First, only two-dimensional data were simulated. Number of dimensions was not selected as a factor to be studied. However, previous research has shown that DETECT is prone to making more classification errors when higher numbers of dimensions are involved in a test (Finch & Habing, 2005; Zhang & Stout, 1999b). Not including number of dimensions as a factor to be studied limits the generalizability of the present study. The pattern of results found in this study might not

be replicated when data with higher numbers of dimensions are simulated. This limitation also reduces the usefulness of the proposed refinement to the evaluation criterion for the D_{max} and r_{max} indices because the new criterion values are only based on results obtained in the two-dimensional case.

Another limitation of the study relates to the item parameters used for simulation. In order to shed light on whether item discrimination parameter has any impact on DETECT classification accuracy and consistency, two sets of item parameters were used to simulate two levels within this factor. Although the SAT parameters involved discrimination parameters as low as 0.2, the maximum value for the discrimination parameters was 1.3. This range overlapped with the LSAT parameters with discrimination parameters of 0.5 to 1.1. Thus, only a portion of the items simulated with the SAT parameters had a low level of discrimination. The overlapping of the two levels of item discrimination also reduces the power of this study to identify differences due to item discrimination parameter. Moreover, the angular departures between the two dimensions for the two sets of parameters were also different, 60° for the LSAT parameter set and 20° for the SAT parameter set. The two factors, overlapping discriminating parameters and differential angular departures, made it impossible to attribute the differences found in the LSAT and the SAT results to one of them.

Implications and Future Directions

Educational and Practical Implications

DETECT, as illustrated in this study, can adequately identify the correct dimensional structure for the two dimensional case classifying items accurately and consistently (at least at the 85% rate) for a limited number of conditions with complex

dimensional structures. The factors that should be considered by researchers and practitioners, when evaluating the trustworthiness of the DETECT results, include degree of complexity in data structure, correlation between dimensions, sample size, and possibly discrimination parameter of items. When tests consisting of items with moderate to high discrimination parameter are analyzed, DETECT can be used even when higher degrees of complexity (e.g., 80% of the items measuring a composite of θ_1 and θ_2) are present, given that the correlation between dimensions is low to moderate (≤ 0.60) and sample size is appropriately chosen (2500 for the correlation of 0.60). When lower degrees of complexity are present in the data, DETECT can be used with confidence if the correlation between dimensions is low to high (≤ 0.80). However, when tests that include items with low discrimination parameter (≤ 0.4) are analyzed, DETECT can only be used with confidence under limited conditions. When data of approximate simple structure are involved, DETECT can be used only when the correlation is low to moderate (≤ 0.60). DETECT will work inaccurately and inconsistently when the data possess complex structure both for the conditions of low (40%) and high (80%) degrees of complexity.

When the nature of the data and the correlation between dimensions are not known to researchers and practitioners, DETECT can still be used, but the interpretation of the results should proceed with caution. With the proposed refinement for the evaluation criterion of the D_{max} and r_{max} index, one can be fairly confident with the classification accuracy of DETECT when a D_{max} value greater than or equal to 0.15 and a r_{max} value greater than or equal to 0.60 are obtained for a dataset.

Future Research Directions

Three major issues still need to be addressed in future studies. First, the present study only investigated the two-dimensional case. How DETECT will perform under conditions of complex structure with higher numbers of dimensions still needs to be studied. This line of research will contribute to our understanding of DETECT under conditions of complex structure and help develop a comprehensive set of guidelines for using and interpreting DETECT results. A simulation study can be conducted setting number of dimensions as a factor to be studied. Different numbers of dimensions (e.g., 3 to 6) can be specified for a simulated dataset. Impact of the factor on $D_{\max}$, $r_{\max}$, classification accuracy, and classification consistency can then be studied.

Second, the two sets of item parameters used in the present study had overlapping ranges of item discrimination parameters. This reduces the power of the study to make a claim about the impact of item discrimination parameter on $D_{\max}$, $r_{\max}$, classification accuracy, and classification consistency. In future research, the use of item parameter sets that have discrete levels of item discrimination will make item discrimination parameter an explicit factor to be studied. Large-scale high-stakes tests as well as low-stakes tests can be sought after to identify item parameter sets with discrete levels of item discrimination.

Moreover, the discrepant results found for the LSAT and SAT parameters could also be attributable to different angular departures between the two dimensions. In the present study, this factor was intertwined with the item discrimination factor, which makes it impossible to separate the effect associated with each of them. In future research, different parameter sets could also be used that have different angular departures between

dimensions to study the effect of angular departure between dimensions on DETECT performance.

Third, the present study was conducted using Kim's (1994) guidelines. How DETECT will perform with the newly proposed guidelines for D_{max} and r_{max} must be addressed. The present study proposed a critical value of 0.15 for the D_{max} index and relaxed the evaluation criterion of the r_{max} index to 0.60. These new guidelines are still tentative and need to be validated systematically through simulation studies of the associated Type I error and power rates. When evaluating the adequacy of the evaluation criterion for D_{max} and r_{max}, unidimensional and multidimensional data should be simulated. The Type I error rate and power associated with the two critical values can then be obtained based on replication data. Only after these studies have been conducted and the correctness of the refinements has been established can we use these new guidelines, with confidence, for directing practice and future research.

References

Ackerman, T. A. (1994). Using multidimensional item response theory to understand what items and tests are measuring. *Applied Measurement in Education, 7*(4), 255-278.

Ackerman, T. A., Gierl, M. J., & Walker, C. M. (2003). Using multidimensional item response theory to evaluate educational and psychological tests. *Educational Measurement: Issues and Practice, 22*(3), 37-53.

Anastasi, A., & Urbina, S. (1996). *Psychological testing* (4th ed.). Upper Saddle River, NJ: Prentice Hall.

Bejar, I. I. (1980). A procedure for investigating the unidimensionality of achievement tests based on item parameter estimates. *Journal of Educational Measurement, 17*(4), 283-296.

Bock, R. D., Gibbons, R., Schilling, S. G., Muraki, E., Wilson, D. T., & Wood, R. (1999). *TESTFACT 3: Test scoring, items statistics, and full-information item factor analysis.* Chicago: Scientific Software International.

Cattell, R. B. (1946). *Description and measurement of personality*. New York: World Book Company.

Chen, W. H. (1993). IRT-LD: A computer program for the detection of pairwise local dependence between test items. (Research Memorandum 93-2). Chapel Hill. University of North Carolina at Chapel Hill, LL. Thurstone Psychometric Laboratory.

Chen, W. H., & Thissen, D. (1997). Local dependence indexes for item pairs using item response theory. *Journal of Educational and Behavioral Statistics, 22*(3), 265-289.

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). Hillsdale, NJ: Lawrence Erlbaum.

Curran, P. J., West, S. G., & Finch, J. F. (1996). The robustness of test statistics to nonnormality and specification error in confirmatory factor analysis. *Psychological Methods, 1*(1), 16-29.

Douglas, J., Kim, H. R., Habing, B., & Gao, F. (1998). Investigating local independence with conditional covariance functions. *Journal of Educational and Behavioral Statistics, 23*(2), 129-151.

Douglas, J., Kim, H. R., Roussos, L., Stout, W., & Zhang, J. (1999). *LSAT Dimensionality Analysis for the December 1991, June 1992, and October 1992 Administrations.* LSAC Research Report Series. Newton, PA: LSAT.

Finch, H. W., & Habing, B. (2005). Comparison of NOHARM and DETECT in item cluster recovery: Counting dimensions and allocating items. *Journal of Educational Measurement, 42*(2), 149-169.

Finch, W. H. (2003). Comparison of the performance of NOHARM and conditional covariance methods of dimensionality assessment: Type I, power and item dimension clustering. (Doctoral dissertation, University of South Carolina). *Dissertation Abstracts International, 64-1A*, 120.

Fraser, C. (1988). *NOHARM: An IBM PC computer program for fitting both unidimensional and multidimensional normal ogive models of latent trait theory.* Armidale, Australia: The University of New England.

Fraser, C., & McDonald, R. P. (1988). NOHARM: Least squares item factor analysis. *Multivariate Behavioral Research, 23*(2), 267-269.

Froelich, A. G., & Habing, B. (2001). Refinements of the DIMTEST methodology for testing unidimensionality and local independence. Paper presented at the annual meeting of the National Council on Measurement in Education, Seattle, WA.

Gierl, M. J. (2005). Using dimensionality-based DIF analyses to identify and interpret constructs that elicit group differences. *Educational Measurement: Issues and Practice, 24*(1), 3-14.

Gierl, M. J., Leighton, J. P., Tan, X. (in press). Evaluating DETECT classification accuracy and consistency when data display complex structure. *Journal of Educational Measurement.*

Gierl, M. J., Tan, X., & Wang, C. (2005). *Identifying content and cognitive dimensions on the SAT.* Research Report No. 2005-11. New York: College Examination Board.

Hakstian, A. R., Rogers, W. T., & Cattell, R. B. (1982). The behavior of number-of-factors rules with simulated data. *Multivariate Behavioral Research, 17*(2), 193-219.

Hambleton, R. K., & Rovinelli, R. J. (1986). Assessing the dimensionality of a set of test items. *Applied Psychological Measurement, 10*(3), 287-302.

Harwell, M., Stone, C. A., Hsu, T., & Kirisci, L. (1996). Monte Carlo Studies in item response theory. *Applied Psychological Measurement, 20*(2), 101-125.

Hattie, J. A. (1985). Methodology review: Assessing unidimensionality of tests and items. *Applied Psychological Measurement, 9*(2), 139-164.

Hattie, J., Krakowski, K., Rogers, H. J., & Swaminathan, H. (1996). An assessment of Stout's index of essential dimensionality. *Applied Psychological Measurement, 20*(1), 1-14.

Holland, P. W., & Rosenbaum, P. R. (1986). Conditional association and unidimensionality in monotone latent variable models. *The Annals of Statistics, 14*(4), 1523-1543.

Jöreskog, K. G., & Sörbom, D. (1993). *LISREL 8: Structural equation modeling with the SIMPLIS command language* [Computer program]. Chicago, IL: Scientific Software, Inc.

Junker, B. W. (1993). Conditional association, essential independence and monotone unidimensional item response models. *The Annals of Statistics, 21*(3), 1359-1378.

Kim, H. R. (1994). New techniques for the dimensionality assessment of standardized test data. (Doctoral dissertation, University of Illinois at Urbana-Champaign). *Dissertation Abstracts International, 55-12B*, 5598.

Kim, H. R., & Stout, W. (1993, April). *A robustness study of ability estimation in the presence of latent trait multidimensionality using the Junker/Stout ε index of dimensionality.* Paper presented at the annual meeting of the American Educational Research Association, Atlanta, GA.

Kuhn, D. (2001). Why development does (and does not) occur: Evidence from the domain of inductive reasoning. In J. L. McClelland & R. Siegler (Eds.), *Mechanisms of cognitive development: Behavioral and neural perspectives* (pp. 221-249). Hillsdale, NJ: Erlbaum.

Leighton, J. P., Gokiert, R. J., & Cui, Y. (in press). Using exploratory and confirmatory methods to identify the cognitive dimensions in large-scale science assessments. *Journal of International Testing.*

Lord, F. M. (1980). *Applications of item response theory to practical testing problems.* Hillsdale, NJ: Lawrence Erlbaum.

Luecht, R. M., Gierl, M. J., & Huff, K. (2006, April). *Scalability and the development of useful diagnostic scales.* Paper presented at the annual meeting of the National Council on Measurement in Education (NCME), San Francisco, CA.

McDonald, R. P. (1967). Nonlinear factor analysis. *Psychometric Monographs,* No. 15.

McDonald, R. P. (1979). The structural analysis of multivariate data: A sketch of general theory. *Multivariate Behavioral Research, 14*(1), 21-38.

McDonald, R. P. (1994). Testing for approximate dimensionality. In D. Laveault, B. D. Zumbo, M. E. Gessaroli, & M. W. Boss (Eds.), *Modern theories of measurement: Problems and issues* (pp. 63-86). Ottawa, Canada: University of Ottawa.

McDonald, R. P. (1997). Normal-ogive multidimensional model. In W. J. van der Linden & R. K. Hambleton (Eds.), *Handbook of modern item response theory.* New York: Springer.

McDonald, R. P. (2000). A basis for multidimensional item response theory. *Applied Psychological Measurement, 24*(2), 99-114.

Miller, T. R., & Hirsch, T. M. (1992). Cluster analysis of angular data in applications of multidimensional item response theory. *Applied Measurement in Education, 5*(3), 193–211.

Mislevy, R. J. (1986). Recent developments in the factor analysis of categorical variables. *Journal of Educational Statistics, 11*(1), 3-31.

Molenaar, I. W., & Sijtsma, K. (2000). *User's manual MSP5 for Windows: A program for Mokken scale analysis for polytomous items* [software manual]. Groningen, The Netherlands: iec ProGAMMA.

Monahan, P. O., Stump, T. E., Finch, H., & Hambleton, R. K. (2005, April). *Bias of exploratory and cross-validated detect index under null hypothesis of unidimensionality.* Paper presented at the annual meeting of the National Council on Measurement in Education, Montreal, QC, Canada.

Mroch, A. A., & Bolt, D. M. (2006). A simulation comparison of parametric and nonparametric dimensionality detection procedures. *Applied Measurement in Education, 19*(1), 67-91.

Nandakumar, R. (1991). Traditional dimensionality versus essential dimensionality. *Journal of Educational Measurement, 28*(2), 99-117.

Nandakumar, R. (1993). Assessing essential unidimensionality of real data. *Applied Psychological Measurement, 17*(1), 29-38.

Nandakumar, R., & Ackerman, T. A. (2004). Test modeling. In K. Kaplan (Ed.), *The Sage handbook of quantitative methodology for the social science* (pp. 93-105). Thousand Oaks, CA: Sage.

Nandakumar, R., & Stout, W. (1993). Refinements of Stout's procedure for assessing latent trait unidimensionality. *Journal of Educational Statistics, 18*(1), 41-68.

Nussbaum, E. M., Hamilton, L. S., Snow, R. E. (1997). Enhancing the validity and usefulness of large-scale educational assessments: IV. NELS:88 science achievement to 12th grade. *American Educational Research Journal, 34*(1), 151-173.

O'Callaghan, R. K., Morley, M. E., & Schwartz, A. (2004, April). *Developing skill categories for the SAT math section.* Paper presented at the annual meeting of the National Council on Measurement in Education, San Diego, CA.

Oh, H., & Reshetar, R. (2004, April). *SAT® I: Reasoning test takers' guessing strategy and their understanding of formula scoring.* Paper presented at the annual meeting of the American Educational Research Association (AERA), San Diego, CA.

Olsson, U. (1979). On the robustness of factor analysis against crude classification of the observations. *Multivariate Behavioral Research, 14*(4), 485-500.

Pedhazur, E. J. (1982). *Multiple regression in behavioral research: Explanation and prediction* (2nd Ed.). New York, NY: Holt, Rinehart, and Winston.

Reckase, M. D. (1979). Unifactor latent trait models applied to multifactor tests: Results and implications. *Journal of Educational Statistics, 4*(3), 207-230.

Reckase, M. D. (1985). The difficulty of test items that measure more than one ability. *Applied Psychological Measurement, 9*(4), 401-412.

Reckase, M. D. (1997). A linear logistic multidimensional model for dichotomous item response data. In W. J. van der Linden & R. K. Hambleton (Eds.), *Handbook of modern item response theory* (pp. 271-286). New York: Springer.

Reckase, M. D., Ackerman, T. A., & Carlson, J. E. (1988). Building unidimensional tests using multidimensional items. *Journal of Educational Measurement, 25*(3), 193-203.

Roussos, L. A. (1993). Hierarchical agglomerative clustering computer program users manual. Unpublished manuscript. Department of Statistics, University of Illinois at Urbana-Champaign.

Roussos, L. A. (1995). *Hierarchical agglomerative clustering computer program user's manual*. Urbana-Champaign: Statistical Laboratory for Educational and Psychological Measurement, Department of Statistics, University of Illinois.

Roussos, L. A., & Ozbek, O. (2003, April). *Formulation of the DETECT population parameter and evaluation of DETECT estimator bias*. Paper presented at the annual meeting of National Council on Measurement in Education (NCME), Chicago, IL.

Roussos, L. A., & Stout, W. F. (1996). Simulation studies of the effects of small sample size and studied item parameters on SIBTEST and Mantel-Haenszel type I error performance. *Journal of Educational Measurement, 33*(2), 215-230.

Roussos, L. A., Stout, W. F., & Marden, J. I. (1998). Using new proximity measures with hierarchical cluster analysis to detect multidimensionality. *Journal of Educational Measurement, 35*(1), 1-30.

Sattler, J. M. (2001). *Assessment of children: Cognitive applications* (4th ed.). San Diego, CA: Jerome M. Sattler Publisher.

Shealy, R. T., Stout, W. F. (1993). A model-based standardization approach that separates true bias/DIF from group differences and detects test bias/DIF as well as item bias/DIF. *Psychometrika, 58*(2), 159-194.

Sijtsma, K., & van der Ark, L. A. (2001). Progress in NIRT analysis of polytomous item scores: Dilemmas and practical solutions. In A. Boomsma, M. A. J. van Duijn, & T. A. B. Snijders (Eds.), *Essays on item response theory* (pp. 297-318). New York: Springer.

Sokal, R. R., & Michener, C. D. (1958). A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin, 38*, 1409-1438.

SPSS 14.0 for Windows. (2005). Chicago, IL: SPSS Inc.

Standards for Educational and Psychological Testing. (1999). Washington, DC: American Educational Research Association, American Psychological Association, National Council on Measurement in Education.

Stout, W. (1987). A nonparametric approach for assessing latent trait unidimensionality. *Psychometrika, 52*(4), 589-617.

Stout, W. (2002). Psychometric: From practice to theory and back. *Psychometrika, 67*(4), 485-518.

Stout, W., Habing, B., Douglas, J., Kim, H. R., Roussos, L., & Zhang, J. (1996). Conditional covariance-based nonparametric multidimensionality assessment. *Applied Psychological Measurement, 20*(4), 331-354.

Sympson, J. B. (1978). A model for testing multidimensional items. In D. J. Weiss (Ed.), *Proceedings of the 1977 Computerized Adaptive Testing Conference* (pp. 82-98). Minneapolis: University of Minnesota, Department of Psychology.

Tate, R. (2002). Test dimensionality. In G. Tindal & T. M. Haladyna (Eds.), *Large-scale assessment programs for all students: Validity, technical adequacy, and implementation.* Mahwah, New Jersey: Lawrence Erlbaum Associates, Publishers.

Tate, R. (2004). Implications of multidimensionality for total score and subscore performance. *Applied Measurement in Education, 17*(2), 89-112.

The William Stout Institute for Measurement. (1993). *MULTISIM: IRT-based educational and psychological measurement software.* Urbana-Champaign, IL: University of Illinois at Urbana-Champaign.

Thurstone, L. L. (1947). *Multiple-factor analysis: A development and expansion of The Vectors of Mind.* Chicago, IL, US: University of Chicago Press.

Traub, R. E., & McLean, L. D. (1985). A survey of university policy makers' preferences and expectations for provincial examinations. *Canadian Journal of Higher Education, 15*(3), 9-21.

Uribe-Zarain, X., Nandakumar, R., & Yu, F. (2005, April). *Application of DIMTEST and DETECT for modeling test data.* Paper presented at the annual meeting of the National Council on Measurement in Education, Montreal, QC, Canada.

Van Abswoude, A. A. H., van der Ark, L. A., & Sijtsma, K. (2004). A comparative study of test data dimensionality assessment procedures under nonparametric IRT models. *Applied Psychological Measurement, 28*(1), 3-24.

VanderVeen, A. (2004, April). *Toward a construct of critical reading for the new SAT.* Paper presented at the annual meeting of the National Council on Measurement in Education, San Diego, CA.

Vosniadou, S., & Brewer, W. F. (1992). Mental models of the earth: A study of conceptual change in childhood. *Cognitive Psychology, 24*(4), 535-585.

Walker, C. M., & Beretvas, S. N. (2003). Comparing multidimensional and unidimensional proficiency classifications: Multidimensional IRT as a diagnostic aid. *Journal of Educational Measurement, 40*(3), 255-275.

Yen, W. M. (1984). Effects of local item dependence on the fit and equating performance of the three-parameter logistic model. *Applied Psychological Measurement, 8*(2), 125-145.

Yen, W. M. (1985). Increasing item complexity: A possible cause of scale shrinkage for unidimensional item response theory. *Psychometrika, 50*(4), 399-410.

Zhang, J. (1997). Some fundamental issues in item response theory with applications. (Doctoral dissertation, University of Illinois at Urbana-Champaign). *Dissertation Abstracts International, 57-11B*, 7272.

Zhang, J., & Stout, W. (1999a). Conditional covariance structure of generalized compensatory multidimensional items. *Psychometrika, 64*(2), 129-152.

Zhang, J., & Stout, W. (1999b). The theoretical DETECT index of dimensionality and its application to approximate simple structure. *Psychometrika, 64*(2), 213-249.

Zhang, Y. O., Yu, F., & Nandakumar, R. (2003, April). *The impact of conditional scores on the performance of DETECT.* Paper presented at the annual meeting of the National Council on Measurement in Education, Chicago, IL.

Zwick, W. R. & Velicer, W. F. (1982). Factors influencing four rules for determining the number of components to retain. *Multivariate Behavioral Research, 17*(2), *253-269.*

Zwick, W. R. & Velicer, W. F. (1986). Comparison of five rules for determining the number of components to retain. *Psychological Bulletin, 99*(3), *432-442.*

Appendix A. Visual Basic Code for Batch Processing of Simulation and DETECT Runs

```
Sub simulation()
    Dim comp(1 To 3) As String, cor(1 To 4) As Single
    Dim sample(1 To 3) As Integer, message(12) As String
    Dim h As Integer, i As Integer, j As Integer, a As Integer, b As
    Integer, c As Integer
    Dim outfile As String, str As String, path As String, datpath As
    String, outpath As String
    Dim RetVal, FindIt As String
    path = "localpath\multisim\"
    datpath = " localpath\data\SAT2\"
    outpath = " localpath\result\SAT2\"
    comp(1) = "simple"
    comp(2) = "complex40"
    comp(3) = "complex80"
    cor(1) = 0.6
    cor(2) = 0.7
    cor(3) = 0.8
    cor(4) = 0.9
    sample(1) = 1500
    sample(2) = 2000
    sample(3) = 2500

    'create directories for data and output results.
    For a = 1 To 1      'complexity: 0, 40, 80
        'needed for original set generation.
        str = datpath & comp(a)
        MkDir str
        str = outpath & comp(a)
        MkDir str
        'end for original set folder set up.

        For b = 1 To 2          'correlation: 0.6, 0.7, 0.8, 0.9
            'needed for original set generation.
            str=datpath & comp(a) & "\cor" & FormatNumber(cor(b),1,vbTrue)
            MkDir str
            str=outpath & comp(a) & "\cor" & FormatNumber(cor(b),1,vbTrue)
            MkDir str
            'end for original set folder set up.

            For c = 1 To 3      'sample size: 1500, 2000, 2500
                'needed for original set generation.
                str = datpath & comp(a) & "\cor" & FormatNumber(cor(b), 1,
                vbTrue) & "\" & sample(c)
                MkDir str
                str = outpath & comp(a) & "\cor" & FormatNumber(cor(b), 1,
                vbTrue) & "\" & sample(c)
                MkDir str
                'end for original set folder set up.

                'needed for cross-validation set generation.
                'str = datpath & comp(a) & "\cor" & FormatNumber(cor(b), 1,
                vbTrue) & "\" & sample(c) & "c"
                'MkDir str
                'str = outpath & comp(a) & "\cor" & FormatNumber(cor(b), 1,
                vbTrue) '& "\" & sample(c) & "c"
                'MkDir str
                'end for cross-valid set folder set up.
```

```
        Next c
    Next b
Next a
'start simulation, MULTISIM syntax generation, same for all conditions.
For h = 1 To 100
    outfile = path & "MUL" & h & ".TXT"
    Open outfile For Output As #1
        Print #1, "SIM" & h & ".TXT"
    Close #1
Next h

For a = 1 To 3              'sample size: 1500, 2000, 2500

    'DETECT syntax generation, same within each sample condition.
    For h = 1 To 100
        outfile = path & "DSYN" & h & ".TXT"
        message(1) = "SIM" & h & ".DAT"
        message(2) = "40"
        message(3) = sample(a)                  'sample size
        message(4) = "10"
        message(5) = "5"
        message(6) = "2"
        message(7) = "DSYN" & h & ".OUT"
        message(8) = "0"
        message(9) = "-1234"
        message(10) = "0"
        Open outfile For Output As #1
            For i = 1 To 10
                Print #1, message(i)
            Next i
        Close #1
    Next h

    For b = 1 To 1          'complexity: 0, 40, 80
        For c = 1 To 2      'correlation: 0.6, 0.7, 0.8, 0.9

            'Multisim simh.txt generation, differ for each condition.
            For h = 1 To 100
            outfile = path & "SIM" & h & ".TXT"
                message(1) = "SIM" & h & ".OUT"
                message(2) = "2"
                message(3) = "40"
                message(4) = sample(a)                      'sample size
                message(5) = "1"
                message(6) = "1"
                message(7) = FormatNumber(cor(c), 1, vbTrue)   'cor
                message(8) = "0.0"
                message(9) = "0.0"
                message(10) = 0 - h - 100    'random seed change for CV
                message(11) = "1"
                message(12) = "SIM" & h & ".DAT"

            Open outfile For Output As #1
                For i = 1 To 3
                    Print #1, message(i)
                Next i
                Open path & "Folder\" & comp(b) & ".prm " For Input As
                #2                          'item parameter file
                    For j = 1 To 40
                        Input #2, str
```

```
            Print #1, str
        Next j
    Close #2
    For i = 4 To 12
        If i = 5 Or i = 8 Then
            For j = 1 To 4
                Print #1, message(i)
            Next j
        Else
            Print #1, message(i)
        End If
    Next i
Close #1
Next h

'run multisim.bat
RetVal = Shell(path & "multisim.bat", 1)
'wait for multisim.bat to finish.
FindIt = Dir(path & "SIM100.DAT")
While Len(FindIt) = 0
    FindIt = Dir(path & "SIM100.DAT")
Wend

'run detectsyn.bat
RetVal = Shell(path & "detectsyn.bat")
'wait for detectsyn.bat to finish.
FindIt = Dir(path & "DSYN100.OUT")
While Len(FindIt) = 0
    FindIt = Dir(path & "DSYN100.OUT")
Wend

'move .dat and .out files.
Open path & "clean.bat" For Output As #1
    'for original set.
    str = "Move Localpath\multisim\SIM*.DAT " & datpath _
    & comp(b) & "\cor" & FormatNumber(cor(c),1,vbTrue) _
    & "\" & Trim(CStr(sample(a)))
    Print #1, str
    str = "Move Localpath\multisim\DSYN*.OUT " & outpath _
    & comp(b) & "\cor" & FormatNumber(cor(c), 1, vbTrue) _
    & "\" & Trim(CStr(sample(a)))
    Print #1, str
    'end for original set

    'for cross-validation set.
    'str = "Move Localpath\multisim\SIM*.DAT " & datpath _
    '& comp(b) & "\cor" & FormatNumber(cor(c), 1, vbTrue) _
    '& "\" & Trim(CStr(sample(a))) & "c"
    'Print #1, str
    'str = "Move Localpath\multisim\DSYN*.OUT " & outpath _
    '& comp(b) & "\cor" & FormatNumber(cor(c), 1, vbTrue) _
    '& "\" & Trim(CStr(sample(a))) & "c"
    'Print #1, str
    'end for cross-valid set.

Close #1
RetVal = Shell(path & "clean.bat")
'wait for clean.bat to finish.
FindIt = Dir(path & "DSYN99.OUT")
While Len(FindIt) <> 0
```

```
                FindIt = Dir(path & "DSYN99.OUT")
            Wend
        Next c
    Next b
Next a

'final clean up of intermediate files.
RetVal = Shell(path & "finalclean.bat")
End Sub
```

Content of "multisim.bat":

```
cd localpath\multisim
MULTISIM < MUL1.txt
…
MULTISIM < MUL100.txt
```

Content of "detectsyn.bat":

```
cd localpath\multisim
DETECT < DSYN1.txt
…
DETECT < DSYN100.txt
```

Content of "finalclean.bat":

```
del Localpath\multisim\SIM*.TXT
del Localpath\multisim\SIM*.OUT
del Localpath\multisim\MUL*.TXT
del Localpath\multisim\DSYN*.TXT
del Localpath\multisim\BRIEF.OUT
```

Appendix B. Visual Basic Code for Batch Processing DETECT Output and Calculating Classification Accuracy

```
Sub addsheet()
    'insert enough worksheets.
    For i = 1 To 34
    Sheets.Add After:=Sheets(i + 2)
    Next i
End Sub
Sub result()
    Dim comp(1 To 3) As String, cor(1 To 4) As Single, sample(1 To 3) As
    Integer, h As Integer, i As Integer, j As Integer, a As Integer, b As
    Integer, c As Integer, str As String, path As String, path1 As String
    Dim k As Integer, e As Integer, d As Integer, correct As Integer, dim1 As
    Integer, correct1 As Integer, correct3 As Integer, f As Integer, g As
    Integer, dim2 As Integer, str2 As String
    path = "Localpath/result/LSAT/"
    comp(1) = "simple  "
    comp(2) = "complex40"
    comp(3) = "complex80"
    cor(1) = 0.6
    cor(2) = 0.7
    cor(3) = 0.8
    cor(4) = 0.9
    sample(1) = 1500
    sample(2) = 2000
    sample(3) = 2500

    'finalresult sheet initialization.
    Sheet37.Name = "finalresult"
    Sheet37.Cells(3, 2) = "DETECTmax"
    Sheet37.Cells(3, 4) = "r Index"
    Sheet37.Cells(3, 6) = "Precision"
    Sheet37.Cells(3, 8) = "Prec I"
    Sheet37.Cells(3, 10) = "Prec II"
    Sheet37.Cells(3, 12) = "Prec Complex"
    For j = 1 To 6
        Sheet37.Cells(4, (j - 1) * 2 + 2) = "Mean"
        Sheet37.Cells(4, (j - 1) * 2 + 3) = "SD"
    Next

    'read DETECT output in and calculate precision.
    For a = 1 To 3              'complexity: 0, 40, 80
        For b = 1 To 4          'correlation: 0.6, 0.7, 0.8, 0.9
            For c = 1 To 3      'sample size: 1500, 2000, 2500

                h = (a - 1) * 12 + (b - 1) * 3 + c      'sheet number

                'read output in.
                path1 = path & Trim(comp(a)) & "/cor" & FormatNumber(cor(b), 1,
               vbTrue) & "/" & Trim(CStr(sample(c))) & "/"
                Worksheets(h).Name = Trim(Left(comp(a), 1) & Right(comp(a), 2))
      & "-" & FormatNumber(cor(b), 1, vbTrue) & "-" & Trim(CStr(sample(c)))
                ThisWorkbook.Worksheets(h).Cells(1, 1) = "Dataset"
                ThisWorkbook.Worksheets(h).Cells(2, 1) = "# of D"
                ThisWorkbook.Worksheets(h).Cells(3, 1) = "Dmax"
                ThisWorkbook.Worksheets(h).Cells(4, 1) = "r Index"
                For i = 1 To 40
                    ThisWorkbook.Worksheets(h).Cells((i + 4), 1) = "Item" & i
                Next i
                ThisWorkbook.Worksheets(h).Cells(45, 1) = "Prec total"
```

```
            ThisWorkbook.Worksheets(h).Cells(46, 1) = "Prec I"
            ThisWorkbook.Worksheets(h).Cells(47, 1) = "Prec II"
            Call readoutput(path1,h)

            'calculate precision.
            Select Case comp(a)
            'Simple precision.
            Case "simple  "
                For i = 1 To 100
                    If i \ 26 = 0 Then
                       str = Chr(65 + i) & "5" & ":" & Chr(65 + i) & "24"
                    Else
                       str = Chr(64 + i\26) & Chr(65+ (i Mod 26)) & "5" &
                       ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "24"
                    End If
                    d =
Application.WorksheetFunction.CountIf(Worksheets(h).Range(str), "=1")
                    e =
Application.WorksheetFunction.CountIf(Worksheets(h).Range(str), "=2")
                    If d >= e Then
                       dim1 = 1
                    Else
                       dim1 = 2
                    End If
                    For j = 1 To 20
                        If Worksheets(h).Cells(j + 4, i + 1) = dim1 Then
                            correct = correct + 1
                            correct1 = correct1 + 1
                        End If
                    Next j
                    For j = 21 To 40
                        If Worksheets(h).Cells(j + 4, i + 1) <> dim1 Then
                            correct = correct + 1
                        End If
                    Next
                    Worksheets(h).Cells(45, i + 1) = correct / 40
                    Worksheets(h).Cells(46, i + 1) = correct1 / 20
                    Worksheets(h).Cells(47, i + 1) = (correct-correct1)/20
                    correct = 0
                    correct1 = 0
                Next i

            'complex40 precision.
            Case "complex40"
                Worksheets(h).Cells(48, 1) = "Prec Complex"
                For i = 1 To 100
                    If i \ 26 = 0 Then
                       str = Chr(65 + i) & "5" & ":" & Chr(65 + i) & "16"
                    Else
                       str = Chr(64 +i\26) & Chr(65 + (i Mod 26)) & "5" &
                       ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "16"
                    End If
                    d =
Application.WorksheetFunction.CountIf(Worksheets(h).Range(str), "=1")
                    e =
Application.WorksheetFunction.CountIf(Worksheets(h).Range(str), "=2")
                    If d >= e Then
                       dim1 = 1
                    Else
                       dim1 = 2
                    End If
                    For j = 1 To 12
                        If Worksheets(h).Cells(j + 4, i + 1) = dim1 Then
```

```
                    correct = correct + 1
                    correct1 = correct1 + 1
                End If
            Next j
            For j = 13 To 24
                If Worksheets(h).Cells(j + 4, i + 1) <> dim1 Then
                    correct = correct + 1
                End If
            Next
            For j = 25 To 32
                If Worksheets(h).Cells(j + 4, i + 1) = dim1 Then
                    correct = correct + 1
                    correct1 = correct1 + 1
                    correct3 = correct3 + 1
                End If
            Next j
            For j = 33 To 40
                If Worksheets(h).Cells(j + 4, i + 1) <> dim1 Then
                    correct = correct + 1
                    correct3 = correct3 + 1
                End If
            Next
            Worksheets(h).Cells(45, i + 1) = correct / 40
            Worksheets(h).Cells(46, i + 1) = correct1 / 20
            Worksheets(h).Cells(47, i + 1) = (correct-correct1)/ 20
            Worksheets(h).Cells(48, i + 1) = correct3 / 16
            correct = 0
            correct1 = 0
            correct3 = 0
        Next i
        Worksheets(h).Cells(48, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B48:CW48"))
        Worksheets(h).Cells(48, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B48:CW48"))

    'complex80 condition.
    Case "complex80"
        Worksheets(h).Cells(48, 1) = "Prec Complex"
        For i = 1 To 100
            If i \ 26 = 0 Then
               str = Chr(65 + i) & "5" & ":" & Chr(65 + i) & "8"
               str2 = Chr(65 + i) & "9" & ":" & Chr(65 + i) & "12"
            Else
               str = Chr(64 +i\26) & Chr(65 + (i Mod 26)) & "5" & _
               ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "8"
               str2 = Chr(64 +i\26) & Chr(65 +(i Mod 26)) & "9" & _
               ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "12"
            End If
            d =
Application.WorksheetFunction.CountIf(Worksheets(h).Range(str), "=1")
            e =
Application.WorksheetFunction.CountIf(Worksheets(h).Range(str), "=2")
            f =
Application.WorksheetFunction.CountIf(Worksheets(h).Range(str2), "=1")
            g =
Application.WorksheetFunction.CountIf(Worksheets(h).Range(str2), "=2")
            If d >= e Then
                If f >= g Then
                    If d >= f Then
                        dim1 = 1
                    Else
                        dim1 = 2
                    End If
```

```
                    Else
                        dim1 = 1
                    End If
                Else
                    If f >= g Then
                        dim1 = 2
                    Else
                        If e >= g Then
                            dim1 = 2
                        Else
                            dim1 = 1
                        End If
                    End If
                End If
                For j = 1 To 4
                    If Worksheets(h).Cells(j + 4, i + 1) = dim1 Then
                        correct = correct + 1
                        correct1 = correct1 + 1
                    End If
                Next j
                For j = 5 To 8
                    If Worksheets(h).Cells(j + 4, i + 1) <> dim1 Then
                        correct = correct + 1
                    End If
                Next
                For j = 9 To 24
                    If Worksheets(h).Cells(j + 4, i + 1) = dim1 Then
                        correct = correct + 1
                        correct1 = correct1 + 1
                        correct3 = correct3 + 1
                    End If
                Next j
                For j = 25 To 40
                    If Worksheets(h).Cells(j + 4, i + 1) <> dim1 Then
                        correct = correct + 1
                        correct3 = correct3 + 1
                    End If
                Next
                Worksheets(h).Cells(45, i + 1) = correct / 40
                Worksheets(h).Cells(46, i + 1) = correct1 / 20
                Worksheets(h).Cells(47, i + 1) = (correct -correct1)/20
                Worksheets(h).Cells(48, i + 1) = correct3 / 32
                correct = 0
                correct1 = 0
                correct3 = 0
            Next i
            Worksheets(h).Cells(48, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B48:CW48"))
            Worksheets(h).Cells(48, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B48:CW48"))

        End Select
        Worksheets(h).Cells(3, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B3:CW3"))
        Worksheets(h).Cells(3, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B3:CW3"))
        Worksheets(h).Cells(4, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B4:CW4"))
        Worksheets(h).Cells(4, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B4:CW4"))
        Worksheets(h).Cells(45, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B45:CW45"))
        Worksheets(h).Cells(45, 103) =
```

```
        Application.WorksheetFunction.StDevP(Worksheets(h).Range("B45:CW45"))
                Worksheets(h).Cells(46, 102) =
        Application.WorksheetFunction.Average(Worksheets(h).Range("B46:CW46"))
                Worksheets(h).Cells(46, 103) =
        Application.WorksheetFunction.StDevP(Worksheets(h).Range("B46:CW46"))
                Worksheets(h).Cells(47, 102) =
        Application.WorksheetFunction.Average(Worksheets(h).Range("B47:CW47"))
                Worksheets(h).Cells(47, 103) =
        Application.WorksheetFunction.StDevP(Worksheets(h).Range("B47:CW47"))

                'write into final result.
                Sheet37.Cells(h + 4, 1) = Worksheets(h).Name
                For k = 1 To 2
                    Sheet37.Cells(h + 4, k + 1) = _
                    ThisWorkbook.Worksheets(h).Cells(3, 101 + k)
                    Sheet37.Cells(h + 4, k + 3) = _
                    ThisWorkbook.Worksheets(h).Cells(4, 101 + k)
                    Sheet37.Cells(h + 4, k + 5) = _
                    ThisWorkbook.Worksheets(h).Cells(45, 101 + k)
                    Sheet37.Cells(h + 4, k + 7) = _
                    ThisWorkbook.Worksheets(h).Cells(46, 101 + k)
                    Sheet37.Cells(h + 4, k + 9) = _
                    ThisWorkbook.Worksheets(h).Cells(47, 101 + k)
                    Sheet37.Cells(h + 4, k + 11) = _
                    ThisWorkbook.Worksheets(h).Cells(48, 101 + k)
                Next k
            Next c
        Next b
    Next a
End Sub

Public Sub readoutput (path1 as String, h as Integer)
      Dim i as integer, j as integer, k as integer, str as string
      For i = 1 To 100
            ThisWorkbook.Worksheets(h).Cells(1, i + 1) = i
            Open path1 & "DSYN" & i & ".OUT" For Input As #1
                  For j = 1 To 8
                        Input #1, str
                  Next j
                  Input #1, str
                  Worksheets(h).Cells(2,i + 1) = Val(Right(Trim(str), 1))
                  For j = 1 To 2
                        For k = 1 To 2
                              Input #1, str
                        Next k
                        Worksheets(h).Cells(2 + j, i + 1) = _
                        Round(Val(Right(Trim(str), 6)), 4)
                  Next j
                  For j = 1 To 6
                        Input #1, str
                  Next j
                  For j = 0 To 3
                        Input #1, str
                        For k = 1 To 10
                              Worksheets(h).Cells(j * 10 + k + 4, i + 1) = _
                              Val(Left(Trim(str), 1))
                              str = Right(Trim(str), Len(Trim(str)) - 1)
                        Next k
                  Next j
            Close #1
      Next i
End Sub
```

Appendix C. Visual Basic Code for Calculating Classification Consistency

```
Sub resultcv()
    Dim i As Integer, cor(1 To 4) As Single
    Dim comp(1 To 3) As String, sample(1 To 3) As Integer
    Dim h As Integer, j As Integer, a As Integer, b As Integer, c As Integer
    Dim str As String, str2 As String, path As String, path1 As String, str3 As
    String, str4 As String, k As Integer, e As Integer, d As Integer, correct As
    Integer, dim1 As Integer, correct1 As Integer, correct3 As Integer,
    realcorrect As Integer, dim2 As Integer, realcorrect1 As Integer,
    realcorrect3 As Integer, f As Integer, g As Integer

    path = "localpath\result\LSAT\"
    comp(1) = "simple  "
    comp(2) = "complex40"
    comp(3) = "complex80"
    cor(1) = 0.6
    cor(2) = 0.7
    cor(3) = 0.8
    cor(4) = 0.9
    sample(1) = 1500
    sample(2) = 2000
    sample(3) = 2500

    'finalresult sheet initialization.
    Sheet37.Cells(41, 2) = "DETECTmax"
    Sheet37.Cells(41, 4) = "r Index"
    Sheet37.Cells(41, 6) = "Matching Rate"
    Sheet37.Cells(41, 8) = "Real Matching Rate"
    Sheet37.Cells(41, 10) = "Matching I"
    Sheet37.Cells(41, 12) = "Real Matching I"
    Sheet37.Cells(41, 14) = "Matching II"
    Sheet37.Cells(41, 16) = "Real Matching II"
    Sheet37.Cells(41, 18) = "Matching Complex"
    Sheet37.Cells(41, 20) = "Real Matching Complex"
    For j = 1 To 10
        Sheet37.Cells(42, (j - 1) * 2 + 2) = "Mean"
        Sheet37.Cells(42, (j - 1) * 2 + 3) = "SD"
    Next

    'read DETECT output in and calculate precision.
    For a = 1 To 3                'complexity: 0, 40, 80
        For b = 1 To 4            'correlation: 0.6, 0.7, 0.8, 0.9
            For c = 1 To 3        'sample size: 1500, 2000, 2500
                h = (a - 1) * 12 + (b - 1) * 3 + c
                'read output in.
                path1 = path & Trim(comp(a)) & "\cor" & FormatNumber(cor(b), 1,
                vbTrue) & "\" & Trim(CStr(sample(c))) & "c\"
                ThisWorkbook.Worksheets(h).Cells(50, 1) = "Dataset"
                ThisWorkbook.Worksheets(h).Cells(51, 1) = "# of D"
                ThisWorkbook.Worksheets(h).Cells(52, 1) = "Dmax"
                ThisWorkbook.Worksheets(h).Cells(53, 1) = "r Index"
                For i = 1 To 40
                    ThisWorkbook.Worksheets(h).Cells((i + 53), 1) = "Item" & i
                Next i
                ThisWorkbook.Worksheets(h).Cells(94, 1) = "Match R"
                ThisWorkbook.Worksheets(h).Cells(95, 1) = "Real MatchR"
                ThisWorkbook.Worksheets(h).Cells(96, 1) = "Match I"
                ThisWorkbook.Worksheets(h).Cells(97, 1) = "Real MatchI"
                ThisWorkbook.Worksheets(h).Cells(98, 1) = "Match II"
                ThisWorkbook.Worksheets(h).Cells(99, 1) = "Real MatchII"
                ThisWorkbook.Worksheets(h).Cells(100, 1) = "Match III" 'complex
```

```
ThisWorkbook.Worksheets(h).Cells(101, 1) = "Real MatchIII"
Call readoutput(path1,h)

'calculate cross-validation.
Select Case comp(a)
'Simple precision.
Case "simple  "
    For i = 1 To 100
        If i \ 26 = 0 Then
           str = Chr(65 + i) & "5" & ":" & Chr(65 + i) & "24"
           str2 = Chr(65 + i) & "54" & ":" & Chr(65 + i) & "73"
           str3 = Chr(65 + i) & "25" & ":" & Chr(65 + i) & "44"
           str4 = Chr(65 + i) & "74" & ":" & Chr(65 + i) & "93"
        Else
           str = Chr(64 +i\26) & Chr(65 + (i Mod 26)) & "5" &
           ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "24"
           str2 = Chr(64+i\26) & Chr(65 +(i Mod 26)) & "54" &
           ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "73"
           str3 = Chr(64+i\26) & Chr(65 +(i Mod 26)) & "25" &
           ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "44"
           str4 = Chr(64+i\26) & Chr(65 +(i Mod 26)) & "74" &
           ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "93"
        End If
        Call judge(h,dim1,dim2,str,str2,str3,str4)
        For j = 1 To 20
            Call simpledim1(dim1,dim2,h,j,correct,realcorrect)
        Next j
        correct1 = correct
        realcorrect1 = realcorrect
        For j = 21 To 40
            Call simpledim2(dim1,dim2,h,j,correct,realcorrect)
        Next
        Worksheets(h).Cells(94, i + 1) = correct / 40
        Worksheets(h).Cells(95, i + 1) = realcorrect / 40
        Worksheets(h).Cells(96, i + 1) = correct1 / 20
        Worksheets(h).Cells(97, i + 1) = realcorrect1 / 20
        Worksheets(h).Cells(98, i+1) = (correct - correct1)/20
        Worksheets(h).Cells(99, i + 1) = (realcorrect - _
        realcorrect1) / 20
        correct = 0
        correct1 = 0
        realcorrect = 0
        realcorrect1 = 0
    Next i

'complex40 precision.
Case "complex40"
    For i = 1 To 100
        If i \ 26 = 0 Then
           str = Chr(65 + i) & "5" & ":" & Chr(65 + i) & "16"
           str2 = Chr(65 + i) & "54" & ":" & Chr(65 + i) & "65"
           str3 = Chr(65 + i) & "17" & ":" & Chr(65 + i) & "28"
           str4 = Chr(65 + i) & "66" & ":" & Chr(65 + i) & "77"
        Else
           str = Chr(64+i\26) & Chr(65 + (i Mod 26)) & "5" &
           ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "16"
           str2 = Chr(64+i\26) & Chr(65 +(i Mod 26)) & "54" &
           ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "65"
           str3 = Chr(64+i\26) & Chr(65 +(i Mod 26)) & "17" &
           ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "28"
           str4 = Chr(64+i\26) & Chr(65 +(i Mod 26)) & "66" &
           ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "77"
        End If
```

```
                    Call judge(h,dim1,dim2,str,str2,str3,str4)
                    For j = 1 To 12
                        Call simpledim1(dim1,dim2,h,j,correct,realcorrect)
                    Next j
                    correct1 = correct
                    realcorrect1 = realcorrect
                    For j = 13 To 24
                        Call simpledim2(dim1,dim2,h,j,correct,realcorrect)
                    Next
                    For j = 25 To 32
                        Call _
                        complexdim1(dim1,dim2,h,j,correct,realcorrect, _
                        correct1, realcorrect1,correct3,realcorrect3)
                    Next j
                    For j = 33 To 40
                        Call _
                        Complexdim2(dim1,dim2,h,j,correct,realcorrect, _
                        correct1, realcorrect1,correct3,realcorrect3)
                    Next j
                    Worksheets(h).Cells(94, i + 1) = correct / 40
                    Worksheets(h).Cells(95, i + 1) = realcorrect / 40
                    Worksheets(h).Cells(96, i + 1) = correct1 / 20
                    Worksheets(h).Cells(97, i + 1) = realcorrect1 / 20
                    Worksheets(h).Cells(98, i + 1) = (correct-correct1)/20
                    Worksheets(h).Cells(99, i + 1) = (realcorrect - _
                    realcorrect1) / 20
                    Worksheets(h).Cells(100, i + 1) = correct3 / 16
                    Worksheets(h).Cells(101, i + 1) = realcorrect3 / 16
                    correct = 0
                    correct1 = 0
                    correct3 = 0
                    realcorrect = 0
                    realcorrect1 = 0
                    realcorrect3 = 0
                Next i
                Worksheets(h).Cells(100, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B100:CW100"))
                Worksheets(h).Cells(100, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B100:CW100"))
                Worksheets(h).Cells(101, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B101:CW101"))
                Worksheets(h).Cells(101, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B101:CW101"))

            'complex80 precision.
            Case "complex80"
                For i = 1 To 100
                    If i \ 26 = 0 Then
                      str = Chr(65 + i) & "5" & ":" & Chr(65 + i) & "8"
                      str2 = Chr(65 + i) & "54" & ":" & Chr(65 + i) & "57"
                      str3 = Chr(65 + i) & "9" & ":" & Chr(65 + i) & "12"
                      str4 = Chr(65 + i) & "58" & ":" & Chr(65 + i) & "61"
                    Else
                      str = Chr(64+i\26) & Chr(65 + (i Mod 26)) & "5" & _
                      ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "8"
                      str2 = Chr(64+i\26) & Chr(65 +(i Mod 26)) & "54" & _
                      ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "57"
                      str3 = Chr(64+i\26) & Chr(65 + (i Mod 26)) & "9" & _
                      ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "12"
                      str4 = Chr(64+i\26) & Chr(65 +(i Mod 26)) & "58" & _
                      ":" & Chr(64 + i \ 26) & Chr(65 + (i Mod 26)) & "61"
                    End If
                    Call judge(h,dim1,dim2,str,str2,str3,str4)
```

```
                    For j = 1 To 4
                        Call simpledim1(dim1,dim2,h,j,correct,realcorrect)
                    Next j
                    correct1 = correct
                    realcorrect1 = realcorrect
                    For j = 5 To 8
                        Call simpledim2(dim1,dim2,h,j,correct,realcorrect)
                    Next
                    For j = 9 To 24
                        Call _
                        complexdim1(dim1,dim2,h,j,correct,realcorrect, _
                        correct1, realcorrect1,correct3,realcorrect3)
                    Next j
                    For j 25 to 40
                        Call _
                        Complexdim2(dim1,dim2,h,j,correct,realcorrect, _
                        correct1, realcorrect1,correct3,realcorrect3)
                    Next j
                    Worksheets(h).Cells(94, i + 1) = correct / 40
                    Worksheets(h).Cells(95, i + 1) = realcorrect / 40
                    Worksheets(h).Cells(96, i + 1) = correct1 / 20
                    Worksheets(h).Cells(97, i + 1) = realcorrect1 / 20
                    Worksheets(h).Cells(98, i + 1) = (correct-correct1)/20
                    Worksheets(h).Cells(99, i + 1) = (realcorrect - _
                    realcorrect1) / 20
                    Worksheets(h).Cells(100, i + 1) = correct3 / 32
                    Worksheets(h).Cells(101, i + 1) = realcorrect3 / 32
                    correct = 0
                    correct1 = 0
                    correct3 = 0
                    realcorrect = 0
                    realcorrect1 = 0
                    realcorrect3 = 0
                Next i
                Worksheets(h).Cells(100, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B100:CW100"))
                Worksheets(h).Cells(100, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B100:CW100"))
                Worksheets(h).Cells(101, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B101:CW101"))
                Worksheets(h).Cells(101, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B101:CW101"))
            End Select
            Worksheets(h).Cells(52, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B52:CW52"))
            Worksheets(h).Cells(52, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B52:CW52"))
            Worksheets(h).Cells(53, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B53:CW53"))
            Worksheets(h).Cells(53, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B53:CW53"))
            Worksheets(h).Cells(94, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B94:CW94"))
            Worksheets(h).Cells(94, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B94:CW94"))
            Worksheets(h).Cells(95, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B95:CW95"))
            Worksheets(h).Cells(95, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B95:CW95"))
            Worksheets(h).Cells(96, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B96:CW96"))
            Worksheets(h).Cells(96, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B96:CW96"))
```

```
                Worksheets(h).Cells(97, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B97:CW97"))
                Worksheets(h).Cells(97, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B97:CW97"))
                Worksheets(h).Cells(98, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B98:CW98"))
                Worksheets(h).Cells(98, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B98:CW98"))
                Worksheets(h).Cells(99, 102) =
Application.WorksheetFunction.Average(Worksheets(h).Range("B99:CW99"))
                Worksheets(h).Cells(99, 103) =
Application.WorksheetFunction.StDevP(Worksheets(h).Range("B99:CW99"))

                'write into final result.
                Sheet37.Cells(h + 42, 1) = Worksheets(h).Name
                For i = 1 To 20 Step 2
                    For k = 1 To 2
                        If i < 4 Then
                            Sheet37.Cells(h + 42, k + i) =
                    ThisWorkbook.Worksheets(h).Cells(52 + (i \ 2), 101 + k)
                        Else
                            Sheet37.Cells(h + 42, k + i) =
                    ThisWorkbook.Worksheets(h).Cells(92 + (i \ 2), 101 + k)
                        End If
                    Next k
                Next i
            Next c
        Next b
    Next a
End Sub

Private Sub Judge (h as integer,byRef dim1 as integer, ByRef dim2 as integer,
str as String, Str2 as String, Str3 as String, Str4 as String)
      Dim d as Integer, e as Integer, f as Integer, g as Integer
      d = Application.WorksheetFunction.CountIf(Worksheets(h).Range(str), "=1")
      e = Application.WorksheetFunction.CountIf(Worksheets(h).Range(str), "=2")
      f = Application.WorksheetFunction.CountIf(Worksheets(h).Range(str3),"=1")
      g = Application.WorksheetFunction.CountIf(Worksheets(h).Range(str3),"=2")
      If d > e Then
            dim1 = 1
      ElseIf d < e Then
            dim1 = 2
      Else
            If f >= g Then
                  dim1 = 2
            Else
                  dim1 = 1
            End If
      End If
      d = Application.WorksheetFunction.CountIf(Worksheets(h).Range(str2),"=1")
      e = Application.WorksheetFunction.CountIf(Worksheets(h).Range(str2),"=2")
      f = Application.WorksheetFunction.CountIf(Worksheets(h).Range(str4),"=1")
      g = Application.WorksheetFunction.CountIf(Worksheets(h).Range(str4),"=2")
      If d > e Then
            dim2 = 1
      ElseIf d < e Then
            dim2 = 2
      Else
            If f >= g Then
                  dim2 = 2
            Else
                  dim2 = 1
            End If
```

```
      End If
End Sub

Private Sub simpledim1(dim1 as Integer, dim2 as Integer, h as Integer, j as
Integer, ByRef correct as Integer, ByRef realcorrect as Integer)
      If dim1 = dim2 Then
            If Worksheets(h).Cells(4 + j, i + 1) = _
            Worksheets(h).Cells(53 + j, i + 1) Then
                  correct = correct + 1
            End If
            If dim1 = 1 Then
                  If Worksheets(h).Cells(4 + j, i + 1) = _
                  Worksheets(h).Cells(53 + j, i + 1) And _
                  Worksheets(h).Cells(4 + j, i + 1) = 1 Then
                        realcorrect = realcorrect + 1
                  End If
            Else
                  If Worksheets(h).Cells(4 + j, i + 1) = _
                  Worksheets(h).Cells(53 + j, i + 1) And _
                  Worksheets(h).Cells(4 + j, i + 1) = 2 Then
                        realcorrect = realcorrect + 1
                  End If
            End If
      Else
            If Worksheets(h).Cells(4 + j, i + 1) <> _
            Worksheets(h).Cells(53 + j, i + 1) Then
                  correct = correct + 1
            End If
            If dim1 = 1 Then
                  If Worksheets(h).Cells(4 + j, i + 1) <> _
                  Worksheets(h).Cells(53 + j, i + 1) And _
                  Worksheets(h).Cells(4 + j, i + 1) = 1 Then
                        realcorrect = realcorrect + 1
                  End If
            Else
                  If Worksheets(h).Cells(4 + j, i + 1) <> _
                  Worksheets(h).Cells(53 + j, i + 1) And _
                  Worksheets(h).Cells(4 + j, i + 1) = 2 Then
                        realcorrect = realcorrect + 1
                  End If
            End If
      End If
End Sub

Private Sub simpledim2(dim1 as Integer, dim2 as Integer, h as Integer, j as
Integer, ByRef correct as Integer, ByRef realcorrect as Integer)
      If dim1 = dim2 Then
            If Worksheets(h).Cells(4 + j, i + 1) = _
            Worksheets(h).Cells(53 + j, i + 1) Then
                  correct = correct + 1
            End If
            If dim1 = 1 Then
                  If Worksheets(h).Cells(4 + j, i + 1) = _
                  Worksheets(h).Cells(53 + j, i + 1) And _
                  Worksheets(h).Cells(4 + j, i + 1) = 2 Then
                        realcorrect = realcorrect + 1
                  End If
            Else
                  If Worksheets(h).Cells(4 + j, i + 1) = _
                  Worksheets(h).Cells(53 + j, i + 1) And _
                  Worksheets(h).Cells(4 + j, i + 1) = 1 Then
                        realcorrect = realcorrect + 1
                  End If
```

```
            End If
      Else
            If Worksheets(h).Cells(4 + j, i + 1) <> _
            Worksheets(h).Cells(53 + j, i + 1) Then
                  correct = correct + 1
            End If
            If dim1 = 1 Then
                  If Worksheets(h).Cells(4 + j, i + 1) <> _
                  Worksheets(h).Cells(53 + j, i + 1) And _
                  Worksheets(h).Cells(4 + j, i + 1) = 2 Then
                        realcorrect = realcorrect + 1
                  End If
            Else
                  If Worksheets(h).Cells(4 + j, i + 1) <> _
                  Worksheets(h).Cells(53 + j, i + 1) And _
                  Worksheets(h).Cells(4 + j, i + 1) = 1 Then
                        realcorrect = realcorrect + 1
                  End If
            End If
      End If
End Sub

Private Sub complexdim1(dim1 as Integer, dim2 as Integer, h as Integer, j as
Integer, ByRef correct as Integer, ByRef realcorrect as Integer, ByRef correct1
as Integer, ByRef realcorrect1 as Integer, ByRef correct3 as Integer, ByRef
realcorrect3 as Integer)
      If dim1 = dim2 Then
            If Worksheets(h).Cells(4 + j, i + 1) = _
            Worksheets(h).Cells(53 + j, i + 1) Then
                  correct = correct + 1
                  correct1 = correct1 + 1
                  correct3 = correct3 + 1
            End If
            If dim1 = 1 Then
                  If Worksheets(h).Cells(4 + j, i + 1) = _
                  Worksheets(h).Cells(53 + j, i + 1) And _
                  Worksheets(h).Cells(4 + j, i + 1) = 1 Then
                        realcorrect = realcorrect + 1
                        realcorrect1 = realcorrect1 + 1
                        realcorrect3 = realcorrect3 + 1
                  End If
            Else
                  If Worksheets(h).Cells(4 + j, i + 1) = _
                  Worksheets(h).Cells(53 + j, i + 1) And _
                  Worksheets(h).Cells(4 + j, i + 1) = 2 Then
                        realcorrect = realcorrect + 1
                        realcorrect1 = realcorrect1 + 1
                        realcorrect3 = realcorrect3 + 1
                  End If
            End If
      Else
            If Worksheets(h).Cells(4 + j, i + 1) <> _
            Worksheets(h).Cells(53 + j, i + 1) Then
                  correct = correct + 1
                  correct1 = correct1 + 1
                  correct3 = correct3 + 1
            End If
            If dim1 = 1 Then
                  If Worksheets(h).Cells(4 + j, i + 1) <> _
                  Worksheets(h).Cells(53 + j, i + 1) And _
                  Worksheets(h).Cells(4 + j, i + 1) = 1 Then
                        realcorrect = realcorrect + 1
                        realcorrect1 = realcorrect1 + 1
```

```
                        realcorrect3 = realcorrect3 + 1
                    End If
                Else
                    If Worksheets(h).Cells(4 + j, i + 1) <> _
                    Worksheets(h).Cells(53 + j, i + 1) And _
                    Worksheets(h).Cells(4 + j, i + 1) = 2 Then
                        realcorrect = realcorrect + 1
                        realcorrect1 = realcorrect1 + 1
                        realcorrect3 = realcorrect3 + 1
                    End If
                End If
        End If
End Sub

Private Sub complexdim2(dim1 as Integer, dim2 as Integer, h as Integer, j as
Integer, ByRef correct as Integer, ByRef realcorrect as Integer, ByRef correct1
as Integer, ByRef realcorrect1 as Integer, ByRef correct3 as Integer, ByRef
realcorrect3 as Integer)
        If dim1 = dim2 Then
            If Worksheets(h).Cells(4 + j, i + 1) = _
            Worksheets(h).Cells(53 + j, i + 1) Then
                correct = correct + 1
                correct3 = correct3 + 1
            End If
            If dim1 = 1 Then
                If Worksheets(h).Cells(4 + j, i + 1) = _
                Worksheets(h).Cells(53 + j, i + 1) And _
                Worksheets(h).Cells(4 + j, i + 1) = 2 Then
                    realcorrect = realcorrect + 1
                    realcorrect3 = realcorrect3 + 1
                End If
            Else
                If Worksheets(h).Cells(4 + j, i + 1) = _
                Worksheets(h).Cells(53 + j, i + 1) And _
                Worksheets(h).Cells(4 + j, i + 1) = 1 Then
                    realcorrect = realcorrect + 1
                    realcorrect3 = realcorrect3 + 1
                End If
            End If
        Else
            If Worksheets(h).Cells(4 + j, i + 1) <> _
            Worksheets(h).Cells(53 + j, i + 1) Then
                correct = correct + 1
                correct3 = correct3 + 1
            End If
            If dim1 = 1 Then
                If Worksheets(h).Cells(4 + j, i + 1) <> _
                Worksheets(h).Cells(53 + j, i + 1) And _
                Worksheets(h).Cells(4 + j, i + 1) = 2 Then
                    realcorrect = realcorrect + 1
                    realcorrect3 = realcorrect3 + 1
                End If
            Else
                If Worksheets(h).Cells(4 + j, i + 1) <> _
                Worksheets(h).Cells(53 + j, i + 1) And _
                Worksheets(h).Cells(4 + j, i + 1) = 1 Then
                    realcorrect = realcorrect + 1
                    realcorrect3 = realcorrect3 + 1
                End If
            End If
        End If
End Sub
```